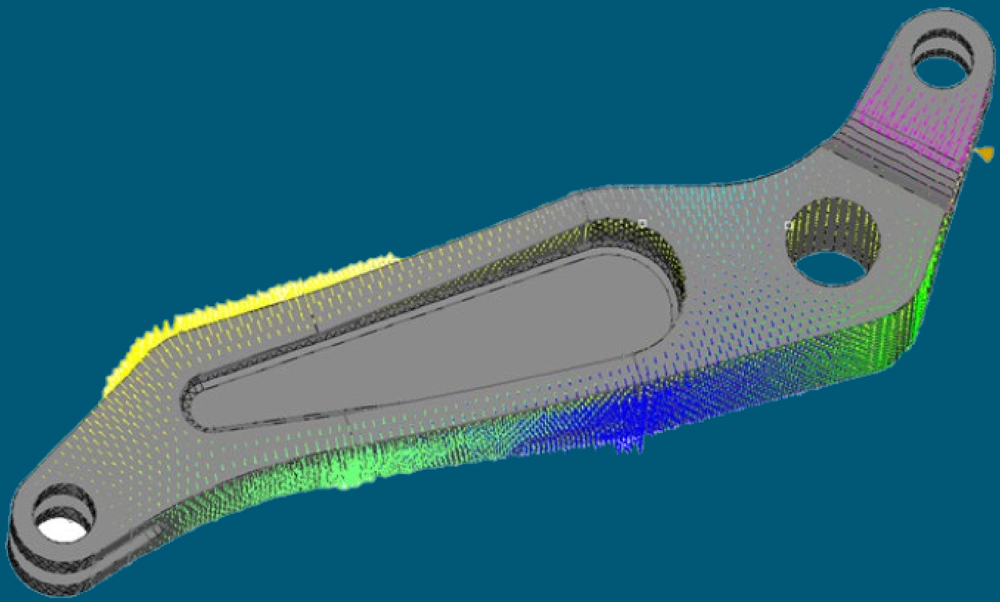
EBOOK

Introduction into Design of Experiments DOE with HyperStudy™



CONTENTS

1 About This Study Guide

Several years back, actually decades, when I was a student myself, I studied parameters (or factors) which may affect the behavior of pre-existing faults (fractures) in the Earth's upper crust. We wanted to understand the "circumstances" which may reactivate a fault (i.e. so that slip along the fault would occur). Some parameters of our investigation were coefficient of friction, orientation and relative magnitude of the mechanical stress field, the dip angle of the fault and so on. These studies have been carried out using the Finite Element Method.

At that time, we created a list of simulation runs considering various combinations of the above parameters. We called it "list of to-do runs". Today we rather would call it simulation run matrix.

Among many other lessons we learned we realized that the manual approach is very prone to errors as every single model variant has to be created, edited and saved manually. Moreover, keeping track of model names, model files, modified parameters/factors, documentation of its effects on fault behavior etc. turned out to be a nightmare – also because every single run lasted several hours.

From the past to present.

Fortunately, some of the technology available today – and in HyperStudy – such as the Hammersley Method helps to overcome (and avoid) the pain points described above.

So why is this old story relevant here?

The objective of this eBook (in fact, interactive PDF) is to demonstrate how to use Altair HyperStudy to perform Design of Experiments (DOE) to solve challenges as described above in a much more efficient way, i.e. to identify critical design variables and their contribution to the design performance more efficiently.

In this eBook, we will describe in some detail, how to perform a DOE study including

- parameterization of models
- defining design variables and responses
- visualization of results, e.g. main effects, interactions, etc.

The content of this book follows quite closely an internal training about Design of Experiments (DOE) given by our colleagues Fatma Koçer and Joseph Pajot (Altair USA). The goal of the training was to bring beginners up to speed with the concepts of DOE and, of course, with our software Altair HyperStudy.

Please note that a commercially released software is a living thing and so at every release (major or point release) new methods, functionalities are added along with improvement to existing methods and functionalities. This document is written using HyperStudy 14.130 except the videos; videos come from a variety of versions some of which are older. As such, there may be some differences between what you see in your screen and what is described here but we certainly think and hope that these differences will not be enough to distract you from using this guide as an effective learning material. If that is not the case, please drop us a note with specifics of distracting discrepancies (or any feedback towards improving the quality of this book). Thank you in advance for your efforts in making this e-book more effective.

We do hope that the selected video recordings from this training as well as some illustrative examples (tutorials) included in the book will help you explore and appreciate the benefits of DOE in the simulation driven product design process.

Let's get started!

Now, before you get started, please recall that regardless of the project complexity you are working on, the reliability of your results is only as good as the person who uses the software: The solver is only capable of working with the problem that you define; it cannot design the part for us and does not have engineering judgment, so the ultimate design decisions lie with YOU!

A very popular statement which summarizes the above (and which is mentioned in many documents) simply says:

Garbage in, Garbage out.

To help you building up simulation experiences (and to overcome some typical roadblocks) we offer additional training & learning resources such as:

- The free HyperWorks Student Edition
- An extended set of E-Learning material (webinars and videos), tutorials about HyperMesh, HyperView, RADIOSS, HyperStudy etc. freely available in the Learning Library http://www.altairuniversity.com/learning-library/
- Additional free study guides about “Practical Aspects of Finite Element Simulation”, “Practical Aspects of Structural Optimization”, “Practical Aspects of Multibody Simulation”, and “Crash Analysis with RADIOSS” (www.altairuniversity.com/ academic program/)
- Learning & Certification Program (http://certification.altairuniversity.com/)
- Highly discounted seminars & workshops at colleges and/or at Altair facilities
- An Altair moderated Support Forum (http://forum.altairhyperworks.com/) and much more

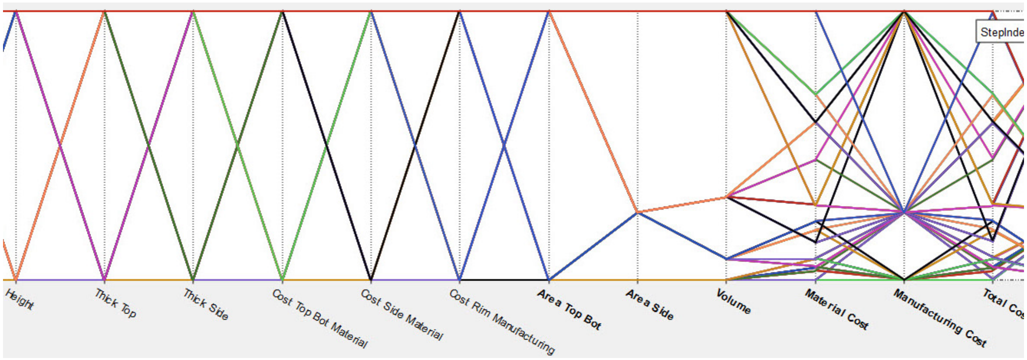
And now - enjoy this study guide and let us know whether DOE (and this book) helped you to successfully solve your design challenges.

Best regards

Dr. Matthias Goelke

On behalf of “The Altair University Team”

Ode to DOE



Three components in a system:

What will happen when I mix ‘em?

I wish I might, I wish I may

Have the answer yesterday.

Outputs changing, lots of action.

Could there be an interaction?

I did my best to weigh it right,

but still an error came to light.

There are those times—I really hate ‘em,

When there’s a problem with a datum.

Some responses I’ve observed,

were modeled best by lines that curved.

Everything’s about to size.

Now’s the time to optimize.

The next assignment that you start,

Work half as long, but twice as smart.

R. C. Neuman

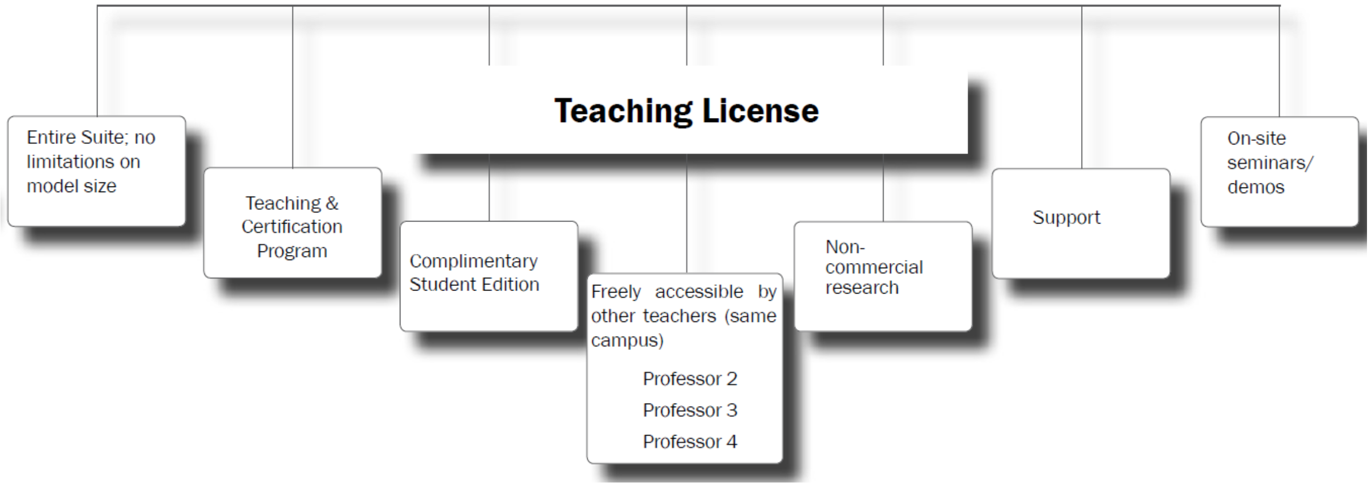
Teaching

Leading universities across the globe are using HyperWorks and SolidThinking computer aided engineering (CAE) simulation software for teaching and research in the fields of:

- Structural Analysis
- Computational Fluid Dynamics (CFD)
- Optimization
- Multi-Body Dynamics/Simulation (MBD/MBS)
- Electro-Magnetic Field Simulation
- Numerical Methods & Programming

Altair has commercial expertise to share with the academic community. By including real life scenarios in your teaching material, Altair can help you add value to your engineering design courses.

Our unique licensing system allows universities to use the entire (full version) HyperWorks & SolidThinking suite in a very flexible and cost-efficient way as sketched out below:



*Results may be used for marketing, training and/demo purposes

Since teaching differs from campus to campus, and from region to region we are very interested to discuss your needs with you personally.

Please let us know your requirements by sending an email notification to altairuniversity@altair.com

We are more than happy helping and assisting you with your teaching activities.

2 Acknowledgement

"If everyone is moving forward together, then success takes care of itself"

Henry Ford (1863 -1947)

A very special **Thank You** goes to:

Fatma Koçer, Joseph Pajot, Franck Delcroix for conducting a very inspirational webinar series and many-many fruitful discussions. Working together with you have been a great learning experience. Thank you for everything.

Ken Mix for providing the interesting video "HyperStudy Overview"

Rahul Rajan for his work on the recordings and review of the manuscript.

Rahul Ponginan, Prakash Pagadala, George Johnson, Sean Putman, Lena Hanna David, Ralph Krawczyk Jr. and Dev Anand for their very valuable support and stimulating discussions.

Lutz Bergmann, Marian Bulla, Christian Buettner, and Vladimir Trickov for all your help and advice.

Diana Mavrudieva for her help reviewing the book and the example of HyperStudy & Flux.

The entire HyperWorks Documentation Team for putting together 1000's of pages of documentation.

Lastly, the entire HyperStudy development team led by Stephan Koerner deserves huge credit for their passion & dedication!

Thank you very much.

3 Disclaimer

4 Simulation Driven Product Design – Fundamentals

4.1 Product Design

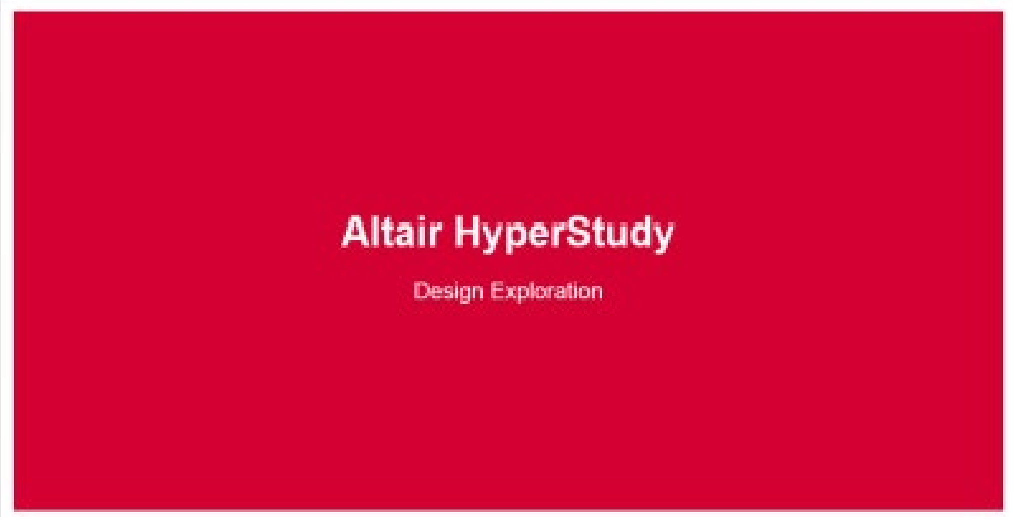
Product design and manufacturing process has been developed over centuries. Many complex systems such as bridges, buildings, aircrafts, cars, etc. are excellent outcomes of this process. This process involves trial-and-error design method in which an initial design is created, tested and modified until a safe design is found.

In most cases, many designs can be safe designs but only one of them is the optimum design which is not only safe but also is the best with respect to a given criterion. Being centuries-old, trial-and-error design method has been developed in the abundance of natural resources and therefore does not directly aim to find the optimum design while it is searching for a safe design. As a result, within the current design process determining the optimum design is an expensive task if not an impossible one as the process does not have a systematic approach to analyzing and comparing different designs.

Let's take a bridge design as an example. Function of a bridge is to facilitate the traffic between two areas that are separated by an obstacle. There are requirements on its safety, cost, durability, maintenance, manufacturability etc. that need to be satisfied. Using the trial-and-error method, engineers cannot evaluate all different possibilities with respect to these requirements and find the optimum bridge design.

As a result, engineers usually adopt a previous design and modify this design until it meets the design requirements. Improvements to increase its value are only made when a new similar design task is started. As a result, evolution of design systems in the past required a lot of time and resources.

With diminishing natural resources, increasing market competition and increasing computational power, trial-and-error method is being replaced by simulation driven design. In our (Altair's) simulation driven design approach, CAD (Computer Aided Design) and CAE (computer aided engineering) are deployed in parallel. Simulation driven design searches not only for a safe design, but it also searches for an optimum design allowing us to reach the best performing design, faster. In other words, the Simulation Driven Design Process eliminates the randomness of the (classical) trial-and-error process.



https://altair-2.wistia.com/medias/uqutjw1t17

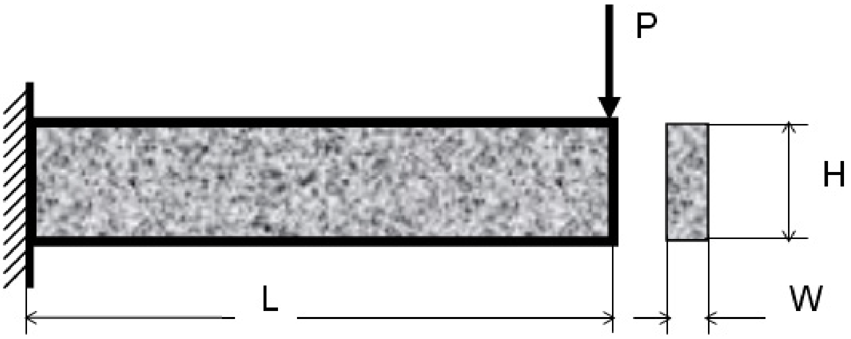
Please note that UI may have slight variations from what you are seeing due to different versions of HyperStudy.

4.2 Design Objectives

Design performance can be defined as how well a product satisfies the design requirements. Design performance is a function of design parameters. In the simulation driven design process, required design performance is divided into design objectives and design constraints.

This is a critical step for the engineers as the outcome will heavily depend on the selection of the objective and constraints.

Let's take a beam design as an example.



This beam is clamped on one end and has a concentrated static load, P, applied on the free end as shown above. We can change the cross-sectional dimensions (H and W), length of the beam (L) and material (r). We can then formulate this problem in several ways.

If we have a defined target on the maximum displacement, then we can identify the objective and constraints as:

- Objective min weight (L, H, W, r)
- Design Constraint Umax (L, H, W, r) ≤ target displacement

If we need to improve its performance but stay also within a target weight, then we can to re-formulate the design problem as:

- Objective min weight (L, H, W, r)
- Design Constraint Umax (L, H, W, r) ≤ target displacement
- Design Constraint weight (L, H, W, r) ≤ target weight.

4.3 Design Process

In the simulation driven design process, ideally as many independent design parameters as possible need to be included since large number of design parameters will create more flexibility in design search. However, in practice, we do not want to spend our computation time on parameters that do not have a large impact on the design. As a result, a Design of Experiment (DOE) for parameter screening is suggested as the first step. Once the problem size is reduced through parameter screening, depending on the problem we can continue with metamodeling. In metamodeling, and fit is created to replace the expensive simulations or to reduce the noise in the data. Then we can perform optimization either on the fit or on the exact model.

For the optimization stage, we can choose to do deterministic studies or probabilistic studies. If we choose the latter, we need to proceed with reliability and robustness-based design optimization (RRBDO).

So, in summary, the simulation driven design process steps are:

- Design of Experiment (DOE) for Screening to reduce problem size.
- DOE for space filling if metamodels are necessary
- Fit (Metamodeling)
- Optimization (Deterministic or probabilistic)
- Stochastic Analysis for Reliability

Before we dive more deeply into Design of Experiments, we will have a brief look at the other phases, namely Fit, Optimization, and Stochastics.

4.3.1 Fit

Fit approach is used to create metamodels that represent the actual responses ("a metamodel or surrogate model is a model of a model, and metamodeling is the process of generating such metamodels" Wikipedia)

They are used to:

- reduce computation time
- smoothen noisy functions so that the optimization algorithms work more effectively.

While working with fits instead of exact simulations, a tradeoff between accuracy and efficiency is ever present. Therefore, it is critical to choose the right DOE method to create the design points as well as the fit method to be used.

Note: Metamodel, Response Surface Model, Surrogate Model, Approximation or Transfer Function are often used as synonyms for Fit.

4.3.2 Optimization

Design optimization is the use of a collection of mathematical procedures that helps the engineers to achieve a design as fully perfect, functional, or effective as possible.

There are three steps in converting a design problem to an optimization problem. These are:

1. Identify design variables:
   - System parameters that can be changed to improve the system performance such as geometrical dimensions, hard-point locations, damping properties etc.
2. Identify objective function(s):
   - System responses that are required to be minimized (maximized) such as mass, displacement, cost etc. These responses should be functions of the design variables.
3. Identify constraint functions:
   - System requirements that need to be satisfied for the design to be feasible such as requirements on stresses, displacements etc. These functions should also be functions of the design variables

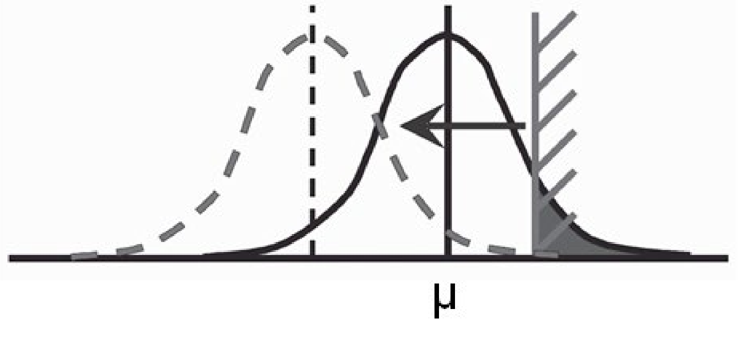
Depending on the nature of the design variables, objective functions and constraints; optimization problems can be classified into different categories such as: constrained problems, discrete problems, multi-objective problems, reliability and robustness based design problems, etc.

In general, there are two types of optimization techniques, iterative and exploratory. Iterative techniques can be either local approximation techniques, or global approximation techniques. Local approximation methods require design sensitivity analysis and are most suitable for linear static, dynamic and multi-body simulations. Global approximation methods are most suitable for nonlinear problems as they are very efficient. Finally, exploratory methods are most suitable for discrete problems and nonlinear simulations, but they are expensive as they require large number of analysis runs.

4.3.3 Stochastics for Reliability and Robustness Based Design

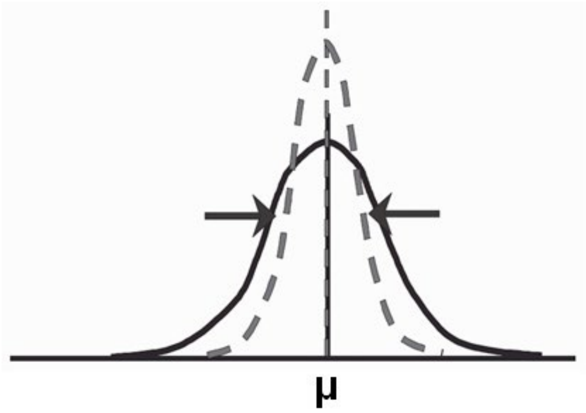
In real world, a design problem description is not deterministic; meaning that there is no one value for the loads imposed, for the material properties used, for the dimensions manufactured but rather there is a range of values with associated variations that are idealized using statistical distributions. Given the probabilistic nature of designs and the fact that optimization drives designs to their limits; it is critical to include reliability and robustness during the design process for problems that are highly sensitive to these variations.

Reliability is the ability of a system to perform and maintain its functions in routine circumstances, as well as unexpected circumstances. Reliability-Based Design Optimization searches for a design that meets the required probability of feasibility given the variations in manufacturing, operations, environment, modeling, etc.



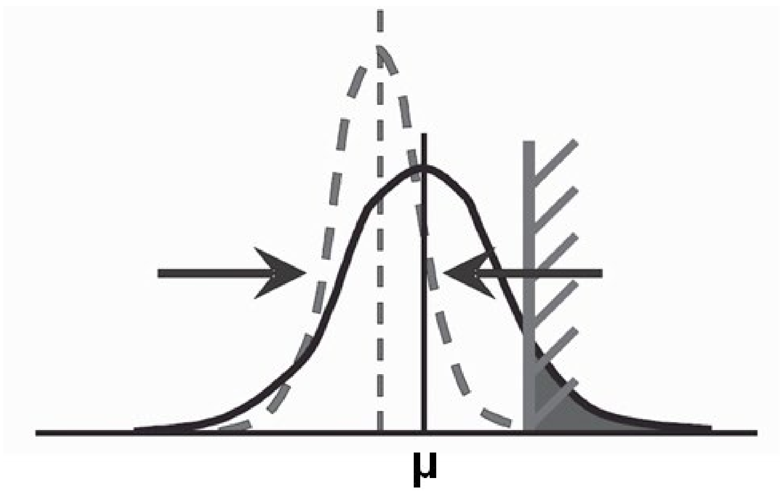
Shifting the “mean of performance” to reduce the failure probability to the required level, i.e. reliability improvement

Robustness is the insensitivity of a design to the same variations as in reliability. In cases where variations in design and operating environment result in the design performance deviation greater than the allowable deviation, it is critical for engineers to search for a robust design. A robust design is one in which the variation in performance due to variation in design is within allowable limits.



Shrinking the "variation of performance" to improve consistency of the product, i.e. robustness improvement

Reliability and robustness-based design optimization covers the requirements of the above two problem formulations as increasing the probability of a feasible design while reducing the sensitivity of the design performances to variations.



Shifting the mean of performance and shrinking the variation of performance, i.e. reliability and robustness improvement

In cases where the design requirements are barely met, reliability may be compromised and could require one to optimize again considering the reliability requirements. Similarly, if variations in design parameters cause high variations in the design performance, search for a robust design becomes necessary. Note, however, that this is often a computationally expensive search.

5 Design of Experiments (DOE)

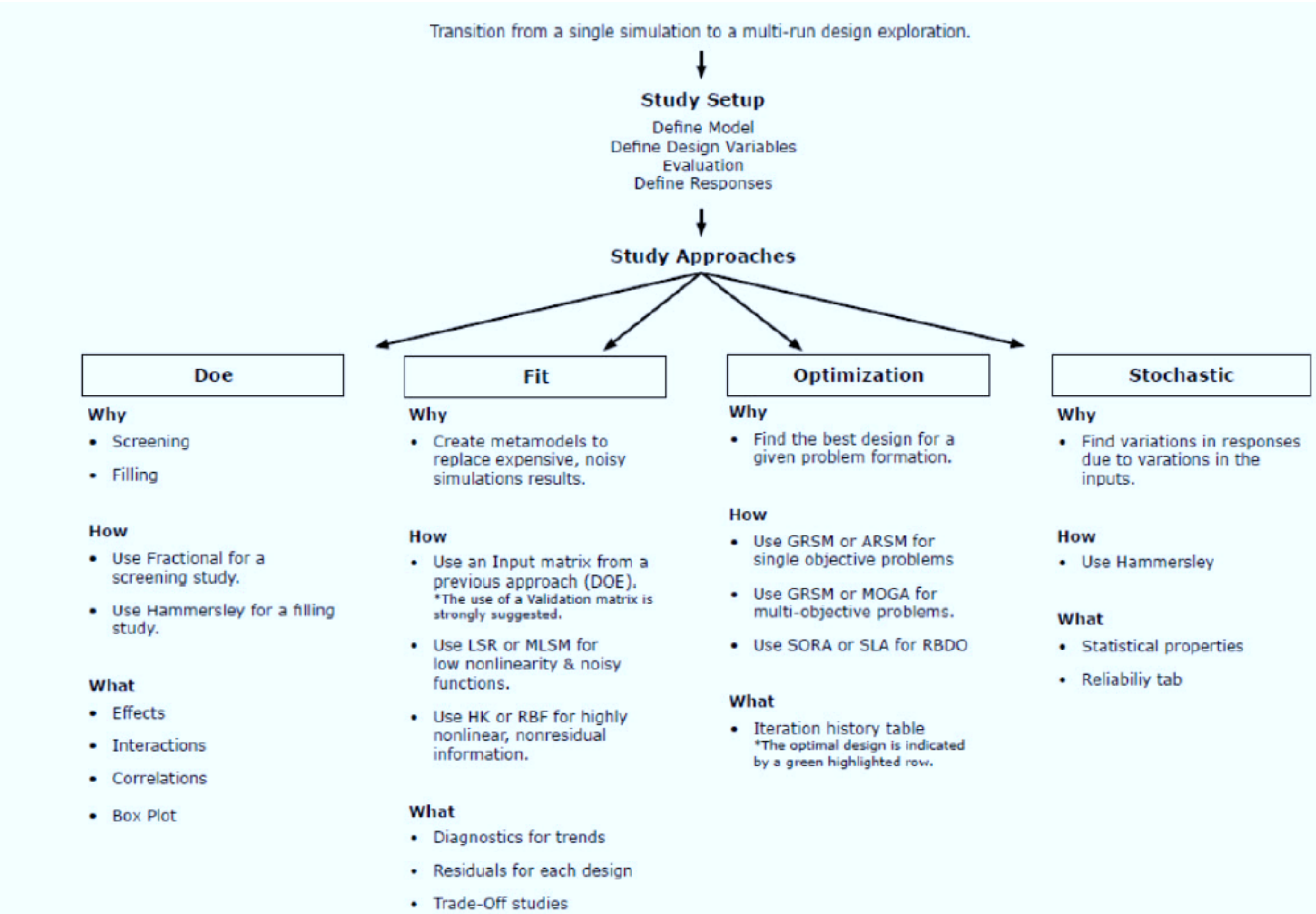
5.1 What is DOE?

Design of Experiments (DOE) can be defined as a series of tests in which purposeful changes are made to the input variables to investigate their effect upon the output responses and to get an understanding of the global behavior of a design problem.

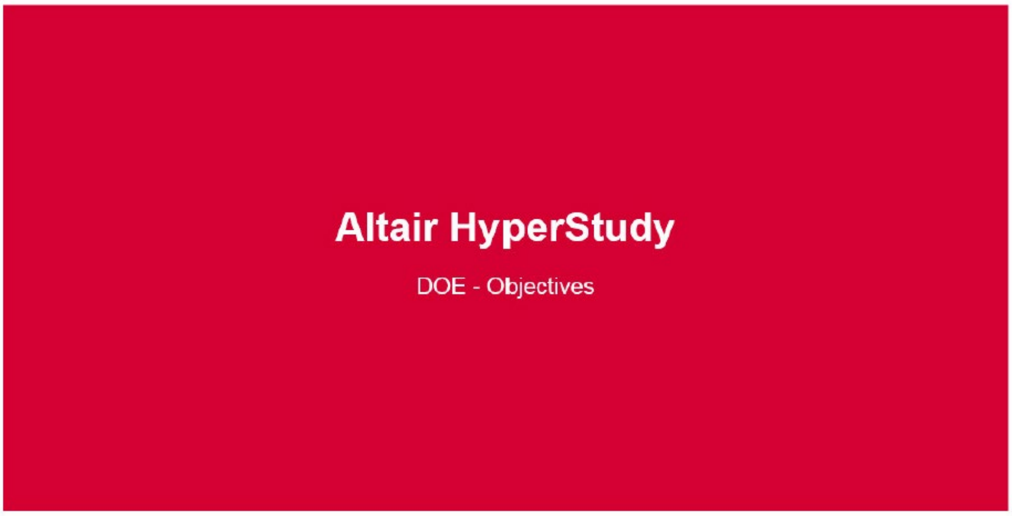
Specifically, it is performed:

- To screen variables by looking for correlation between factors and responses, usually by running a factorial / reduced experiment for a large number of factors with a few levels (as low as 2) for each.
- To construct a fit for an optimization or a stochastic study (to be computationally more effective a fit is used instead of the exact solver).

In general, DOE study runs are independent of each other, so large-scale DOE studies can be run in parallel using parallel processing techniques or using distributed processing techniques with load sharing clusters.

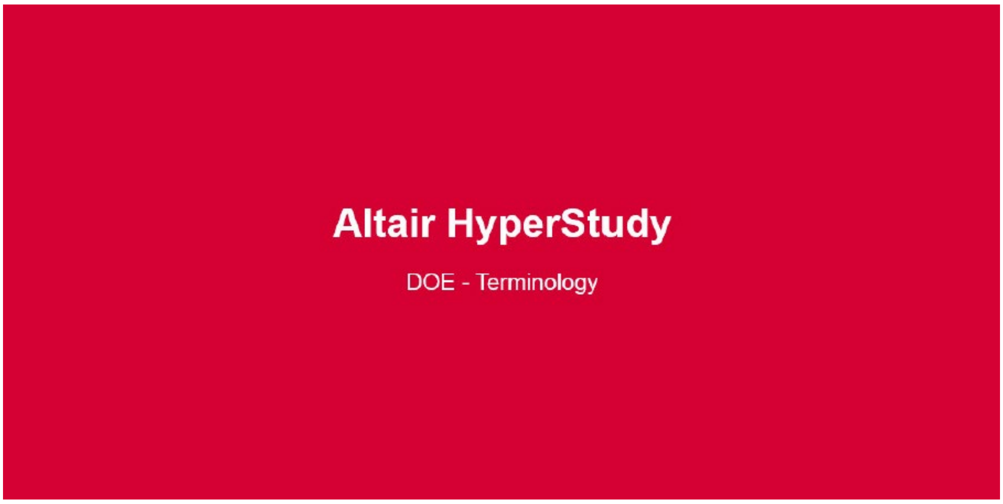


Use the map above to guide during the study setup process. Please be aware that this map does not outline every HyperStudy use case.



https://altair-2.wistia.com/medias/2b32bypi7l

5.2 DOE Terminology



https://altair-2.wistia.com/medias/h37fln2idz

Design of Experiments (DOE)

Is a structured, organized design/run matrix creation method that once run will be used to determine the relationships between the different factors/variables (Xs) affecting a process and the output of that process (Y).

Factors

Are system parameters (or design variables) that change the system performance. Factors can be controlled or uncontrolled.

A factor can be either Discrete i.e., slow (-) or fast (+) or Continuous i.e. temperature.

Variables

Are system parameters that can be changed to improve the system performance. Since this chapter is part of a design study process, we will be using factors and variables interchangeably.

Levels

Are discrete (or continuous) values of the factor/variable. The values taken by in the range [-1; +1]. Number of levels per variable to be considered depends on the level of non-linearity in the problem; for a linear model two levels are sufficient; for a quadratic model three levels are needed.

Controlled factors

Are design variables that can be realistically controlled in the production (real world) environment. In HyperStudy, these are assigned to a distribution role of Design or Design with Random; depending on whether they are deterministic or probabilistic variables. 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. In HyperStudy, these are assigned to a distribution role of Random parameter. Examples include ambient temperature and occupant seating positioning.

Confounding

Is the inability to distinguish between main effects and interaction effects. Confounding occurs when a fractional design is chosen instead of a full-factorial design. The consequence of confounding is that the values calculated for main effects will have error coming from inclusion of higher order interactions in the calculation and interaction effects will be unknown. The amount of confounding in a DOE is quantified by its resolution.

5.3 History

An interesting “story” about a systematic clinical trial to compare remedies for scurvy which constitutes a type of DOE is depicted on Wikipedia (https://en.wikipedia.org/wiki/Design_of_experiments)

In 1747, while serving as surgeon on HMS Salisbury, James Lind carried out a systematic clinical trial to compare remedies for scurvy. This systematic clinical trial

Lind selected 12 men from the ship, all suffering from scurvy. Lind limited his subjects to men who "were as similar as I could have them," that is, he provided strict entry requirements to reduce extraneous variation. He divided them into six pairs, giving each pair different supplements to their basic diet for two weeks. The treatments were all remedies that had been proposed:

- A quart of cider every day.
- Twenty-five guts (drops) of vitriol (sulphuric acid) three times a day upon an empty stomach.
- One half-pint of seawater every day.
- A mixture of garlic, mustard, and horseradish in a lump the size of a nutmeg.
- Two spoonsful of vinegar three times a day.
- Two oranges and one lemon every day.

The citrus treatment stopped after six days when they ran out of fruit, but by that time one sailor was fit for duty while the other had almost recovered. Apart from that, only group one (cider) showed some effect of its treatment. The remainder of the crew presumably served as a control, but Lind did not report results from any control (untreated) group.

As we are looking at the history, one of course, needs to mention the work by Ronald Fisher (1927):

A methodology for designing experiments was proposed by Ronald Fisher, in his innovative books: The Arrangement of Field Experiments (1926) and The Design of Experiments (1935). Much of his pioneering work dealt with agricultural applications of statistical methods. As a mundane example, he described how to test the lady tasting tea hypothesis, that a certain lady could distinguish by flavor alone whether the milk or the tea was first placed in the cup.

Thus, DOE started in testing and therefore classical DOE methods deal with testing behaviors and issues. Since then many industries, like pharmaceutical, chemical, and manufacturing, have successfully applied DOE studies to their product development process.

Lately, as DOE started to be adapted by CAE as an effective way of design exploration, more CAE-oriented methods such as Latin Hypercube and Hammersley (see discussion below for details) have been developed. In a CAE model, for instance a sheet metal forming model, factors such as thickness, shape design variables, and material properties can be changed to study their effects on the model responses, and their respective interactions (if present at all).

For a casual reading on design of experiments versus design optimization, you can read Joe Pajot's blog on

http://innovationintelligence.com/design-exploration-vs-design-optimization/

5.4 Inspirational and Illustrative Example of a DOE

The following example about a Bond Strength Study provides some good insight into DOE and its "power". In this chapter we also explain some basic principles of DOE.

In this project the following assumptions were made:

- Temperature, pressure, and adhesive viscosity affect bond strength
- We have some control over temperature, pressure (controlled factors)
- We do not have control over adhesive viscosity (uncontrolled factor)
- Temperature changes between 70-210 degrees

- Pressure changes between 30-45 psi
- Bond strength is assumed to be a linear function of both temperature and strength within these ranges of temperature and pressure
- As a result, a two-level DOE is enough to study the effects of temperature and pressure on strength

We know that temperature, pressure, and adhesive viscosity affects the bond strength and we need to figure out which temperature and pressure settings produce the highest and most robust bond strength.

In our lab we are able to control pressure and temperature but not viscosity. Viscosity is the property of the adhesive material which is already purchased and so we do not have control over its settings. Since we can’t control viscosity setting (which has an effect on the strength or performance), this variable is called an uncontrolled factor. Pressure and temperature are consequently called controlled factors. Hence, we are going to focus in this study on the factors we can control, temperature and pressure. Of course, this makes our lives easier …

The temperature is assumed to vary between 70 and 210 degrees, the pressure is assumed to vary between 30 and 45 psi.

Further, we assume that the bond strength is a linear function of both, temperature and pressure.

This may also be expressed as a study consisting of two variables, or factors (i.e. Temperature & Pressure). Their values may be set to a lower and upper bound (lower and upper level), respectively. Hence, we are looking at a 2 factor 2 level experiment/ study. This is also called a two-level Design of Experiment.

5.4.1 Effects

To understand the effects of the factors and the respective level we assume the following set of experiments (or runs):

Table of experiments

Experiment	Temperature (degree)	Pressure (psi)
1	70 (level – 1)	30 (level – 1)
2	70 (level – 1)	45 (level – 1)
3	210 (level – 1)	30 (level – 1)
4	210 (level – 1)	45 (level – 1)

In the first experiment temperature and pressure are at their lower bounds: 70 degrees and 30 psi. The lower bounds are named Level -1. Note, Level 1 and Level -1 may also be named level 0 and 1.

In the second experiment the pressure is set to its upper bound, 45 psi which corresponds to level 1. The temperature is kept at its lower value 70 degrees (level -1). This experiment will give us an understanding on how the pressure change (increase) affects the bond strength while keeping the temperature at its lower value.

In the third experiment the temperature is raised to its upper bound 210 degrees (level 1) and the pressure is set to level -1 (lower value of 30 psi)

And finally in experiment 4, both factors are set to their higher bounds, level 1, respectively.

Thus, experiment 1 and 3 will give an understanding on how the temperature change (while keeping the pressure at its lower level) is impacting the bond strength.

With experiments 2 and 4 we look at the effects of temperature increase (set to level 1) while the pressure is at its upper bound (level 1).

The results of these experiments will tell us how the temperature (factor), how the pressure (factor) as well as a combination of temperature and pressure change affect the strength of the bond. In other words, we study the effects of the factors as well as their interaction, meaning how they influence each other.

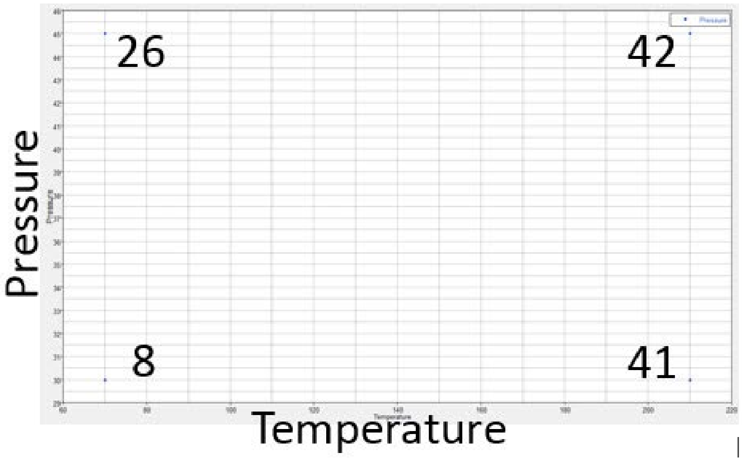
In the image below the strength values as determined through the experiments are listed.

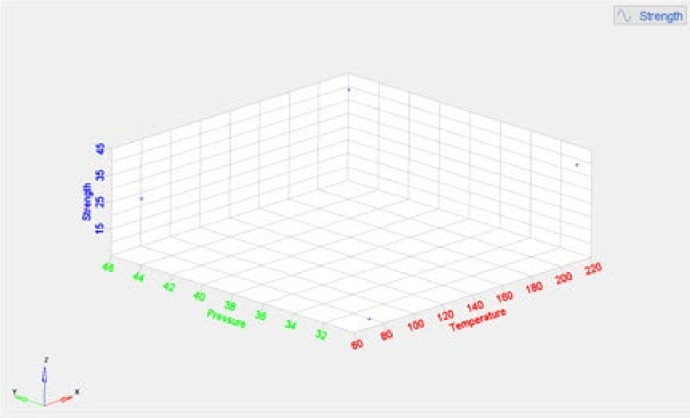
Experiment	Temperature (degree)	Pressure (psi)	Strength (lbs)
1	70.000000	30.000000	8.0000000
2	70.000000	45.000000	26.000000
3	210.00000	30.000000	41.000000
4	210.00000	45.000000	42.000000

Measured bond strength values

In Experiment 1 where both variables (factors) where at their lower bounds (level -1) the strength is 8 lbs. In Experiment 4, with both factors at their upper bound (level 1) the strength is 42 lbs. and so on.

The same results may be shown in a Scatter Plot (taken from HyperStudy)

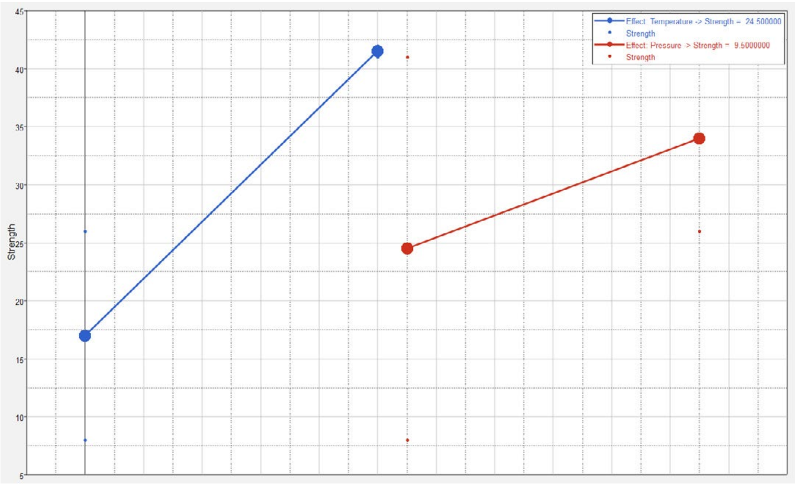




Measured bond strength (with respect to changes in temperature and pressure)

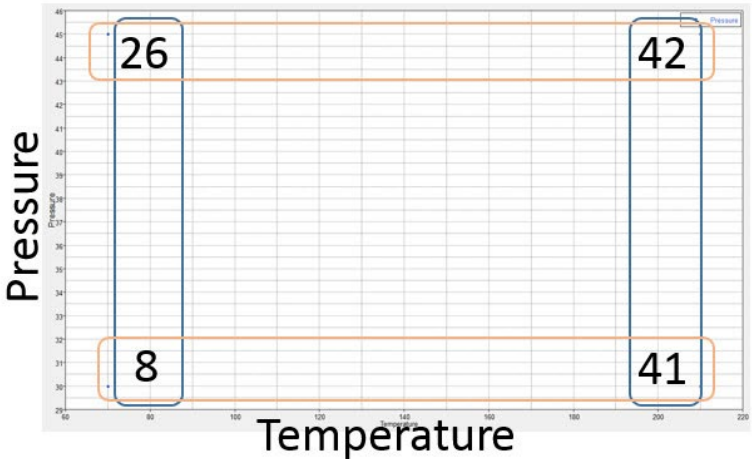
In the lower left corner of the plot, the temperature and pressure is at its lower bound (level -1), in the upper right corner temperature and pressure are at their upper bounds (level 1), etc.

In the next step the effects of each variable/factor and their interaction (with each other) on the strength is plotted.



Main effects plot

The above chart is called "Effects Plot". But how do we get these charts?



First, we try to understand the effect of temperature on its own.

The effect of temperature on strength (framed in dark color in the image above) is determined by averaging the strength values when temperature is high and subtracting the averaged (mean) strength when the temperature is low.

Effect of Temperature on strength:

(42+41)/2 – (26+8)/2 = 24.5 lbs.

The same math is done to understand the effect of pressure on strength (marked in light green):

That is mean value of strength when pressure is high – minus mean value of strength when pressure is low:

Effect of Pressure on strength:

(42+26)/2 – (41+8)/2 = 9.5 lbs.

So, what does these numbers tell us at all?

Look at the temperature plotted in the upper chart. Start point is level -1 (lower temperature bound) with a mean value of 17.0 lbs. At level 1 (upper temperature bound) the mean value is 41.5 lbs. The difference (gradient) between upper and lower level is 24.5 lbs.

We redo the same considerations regarding the pressure effect. Lower bound is 24.5 lbs., upper bound 34 lbs. The difference between upper and lower level is 9.5 lbs.

As reflected by the gradient of the graphs, the temperature (higher gradient) has a greater impact on strength than pressure.

5.4.2 Interactions

In the chart below the x-axis refers to pressure, ranging from 30 to 45 psi. The y-axis is strength ranging from 8 to 42 lbs.

At 70 degrees:

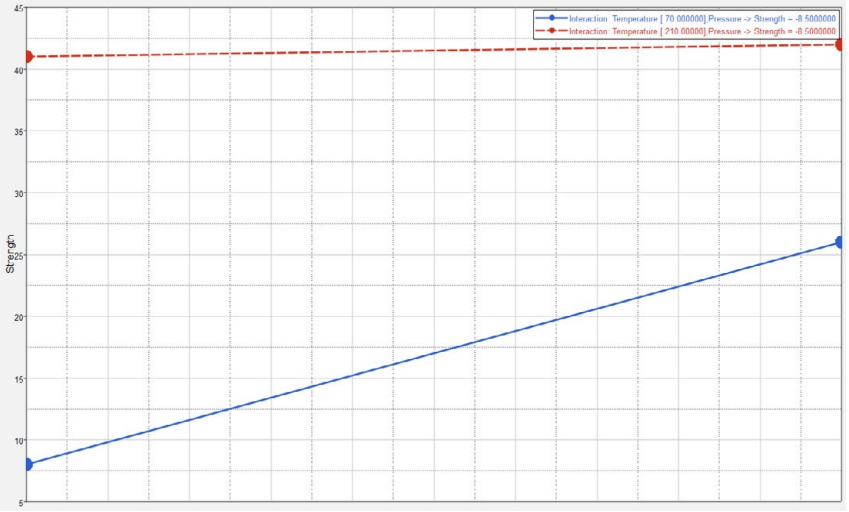
Level 1 - lower left corner pressure 8 lbs.

Level 2 – upper right corner pressure 26 lbs.

At 210 degrees:

Level 1 - lower left corner pressure 41 lbs.

Level 2 – upper right corner pressure 42 lbs



Interaction plot

So, we plot the effect of pressure on the strength at a constant temperature of 70 degrees and 210 degrees, respectively.

As the plotted lines in the chart are not parallel, it tells us that there is an interaction between the factors pressure and temperature. In other words, if both curves were parallel then the effect of temperature on strength would always be the same, regardless of the pressure.

To see (from a more numerical point of view) whether the curves are parallel or not, just determine the difference of the gradients of both curves (gradient of red curve – gradient blue curve):

Effect of Pressure when Temperature=210 is (42-41)/2=0.5

Effect of Pressure when Temperature=70 is (26-8)/2 = 9

Interaction of Temperature on Pressure is = (0.5-9)/2=-8.5

Quite obvious the effect of pressure on strength at a constant temperature of 70 degrees is bigger (or more important) than at higher temperatures.

Moreover, we can also derive from the chart that at low pressure and low temperature the interaction between both variables/ factors is most pronounced, i.e. for lower pressure and lower temperature we get the most negative interaction between both variables/factors.

In contrast, looking at the upper red curve, the gradient is quite small indicating a rather robust situation/relationship between temperature and pressure, and hence resulting in a better bond!

Question: But what does this finding tell me at all? What do we need to do to get the most robust bond strength?

Answer: If we have the budget to control one factor, then we should concentrate on Pressure!?

Sure? Wrong answer!

We should concentrate on temperature!

Thus, if the goal is to have a robust and high strength bond, then we should aim for a temperature of 210 degrees in combination with high pressure (upper right corner of the chart).

Quite obviously, this was a rather small size experiment with two variables (factors) with two levels (lower and higher settings). As seen in the table above, 4 experiments where needed to investigate all different combinations:

Level factor = 22 = 4

In most design projects, however, we are confronted with many more variables. The number of designs (experiments) that cover all possible combinations of design variables and levels

\# of designs = (2# of DVs with 2 levels) (3 # of DVs with 3 levels) (4 # of DVs with 4 levels)

Scary enough, the number of designs grows very fast with the number of design variables.

For instance,

7 designs variables; 2 at 2 levels, 4 at 3 levels, 1 at 4 level

\# of designs = 223441 = 1296 runs

This result immediately provokes the question: Do we have the time, capacity, budget etc. to run all these 1296 experiments?

What if we pick a smart affordable subset of the full set (=1296), can we still capture enough information?

But how should we select the subset then?

This challenge has led to the development of a broad range of Design of Experiment methods as we always aim to maximize the information about the system while investing minimum effort! (sounds familiar as this "wishful thinking" applies not only to product design).

When applying DOE in our CAE projects the latter situation, i.e. many design variables, in combination with limited budget is quite common

To summarize the above introduction

DOE is applied (and strongly recommended):

- To determine which factors are most influential on the responses.
- To construct an approximate model that can be used as a surrogate model for the actual computationally intensive solver.

5.4.3 Confounding

Confounding occurs when two factors are associated with each other or "travel together" and the effect of one is confused with the effect of the other. For example, in order to improve team performance, a soccer coach asks his team to run two miles a day while the players decide to take vitamins. In this case the effects of running two miles a day and taking vitamins will be confounded since it will not be possible to identify the effect of them on team performance independently.

In other words, confounding is the inability to distinguish between main effects and interaction effects.

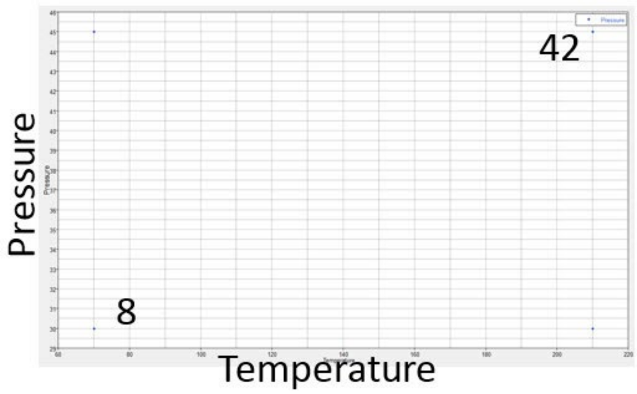
Confounding occurs when a fractional design is chosen instead of a full-factorial design.

The consequence of confounding is that the values calculated for main effects will have error coming from inclusion of higher order interactions in the calculation and interaction effects will be unknown.

To better illustrate this situation, we are looking at the example from the introduction again:

Experiment	Temperature (degree)	Pressure (psi)	Strength (lbs)
1	70.000000	30.000000	8.0000000
2	70.000000	45.000000	
3	210.00000	30.00000	
4	210.00000	45.000000	42.000000

Some experiments are purposely excluded from the study



Assuming that we can't afford to run 4 experiments (as we did before) but only 2. We then may would (rather arbitrarily) delete/ remove/neglect experiment 2 and 3.

Then the effect of pressure and/temperature on the bond strength would be:

Effect of Temperature on strength:

(42) – (8) = 34 lbs.

Effect of Pressure on strength:

(42) – (8) = 34 lbs.

By neglecting the other 2 experiments we confounded the effects of temperature and pressure because we can't distinguish their relative importance anymore, i.e. we can't tell/see the effects of pressure and temperature on the strength independently.

Let's look at the consequences if we would select two other experiments: Experiment 1 and 2 are neglected. Obviously, this selection of experiments allows us to investigate the effects of pressure on the bond strength only; because the experiments are conducted at a constant temperature of 100 degrees.

Experiment	Temperature (degree)	Pressure (psi)	Strength (lbs)
1	70.000000	30.000000	8.0000000
2	70.000000	45.000000	26.000000
3	210.00000	30.000000	
4	210.00000	45.000000	

Effect of Temperature on strength:

0 lbs.

Effect of Pressure on strength:

(26) – (8) = 18 lbs.

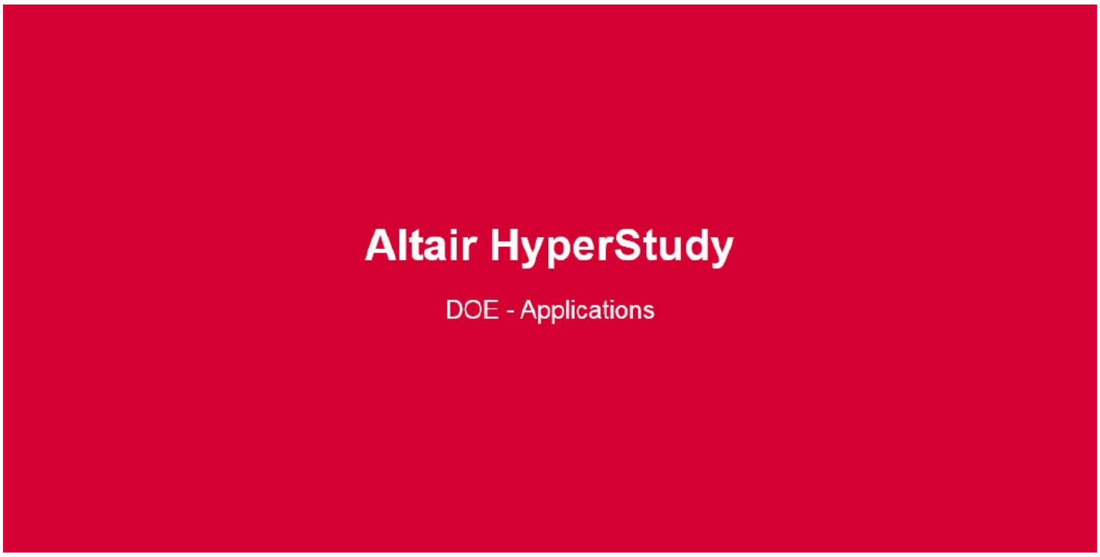
The effect of temperature on strength is zero because we didn't investigate the effect of temperature.

What did we learn from the above?

We started with a very detailed design (4 runs, full factorial), then to 2 runs (pressure and temperature) which gave some indication on their effects on the results (strength) to the last study in which temperature was constant. The latter study, however, yields the least accurate results, i.e. a misleading result.

Hence, picking the correct set of experiments out of a larger number is very important. The good thing about it: You are not left alone with this – HyperStudy helps you defining/selecting the right experiments!

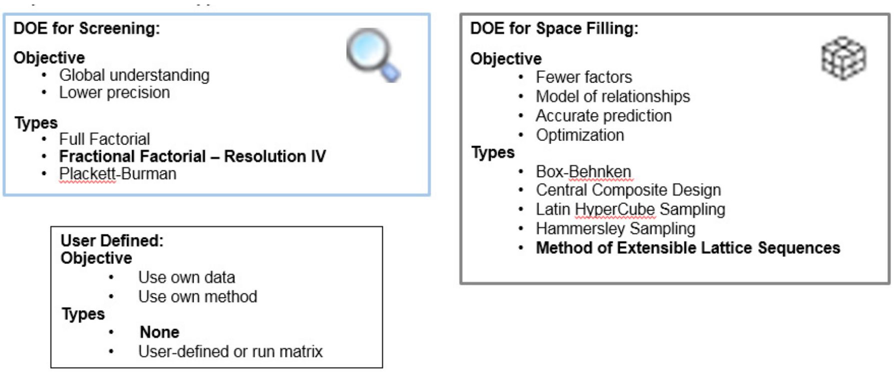
5.5 DOE Study Types



https://altair-2.wistia.com/medias/mbcr7voc3r

DOE studies can be classified according to their functional applications:

- DOE for Screening
- DOE for Space Filling (more used for CAE specific applications)
- User Defined



DOE study types and their typical applications/use cases

5.5.1 DOE for Screening

A DOE study for screening (a two-level design with no interactions, for example) will provide a global understanding of the complete system (giving the magnitude and direction of effects). This initial screening exercise will allow parameters not influencing the system to be discarded, thereby reducing the number of factors and runs. The drawback of a screening DOE is that it is of lower precision.

Types: Full Factorial

Fractional Factorial (takes subsets of full factorial)

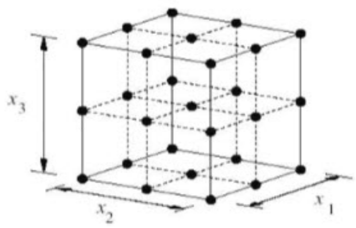
Plackett-Burman

5.5.2 Full Factorial - Screening DOE

A full factorial DOE evaluates responses for all combinations of design variable levels. This will tell us all the effects and interactions but for cases with large number of design variables and levels, the total number of runs becomes large.

In other words, this method may be practical for cases where there is a small number of factors (e.g. 3 factors) and each factor has two levels such as yes or no; -1 or 1 (X1 varies between -1/1; X2: -1/1; X3: -1/1)

Full factorial is not practical for most CAE applications where there are many factors possibly at several levels and the simulations are costly.



Full factorial study with 3 variables with 3 levels

Full Factorial Study (i.e. create all the combinations) with 3 variables with 3 levels ends up in 33 = 27 design runs (depicted by the bold nodes in the image)

Usability Characteristics

- For a case with k design variables, each at L levels, a full factorial design has Lk runs. For studies with a large number of design variables and levels, the total number of runs is larger. For example, for a study with eight factors and each with three levels, 6561 runs are needed (38 = 6561).
- If the number of levels is not equal across variables, then the total number of runs is calculated by the product of the Lk terms.

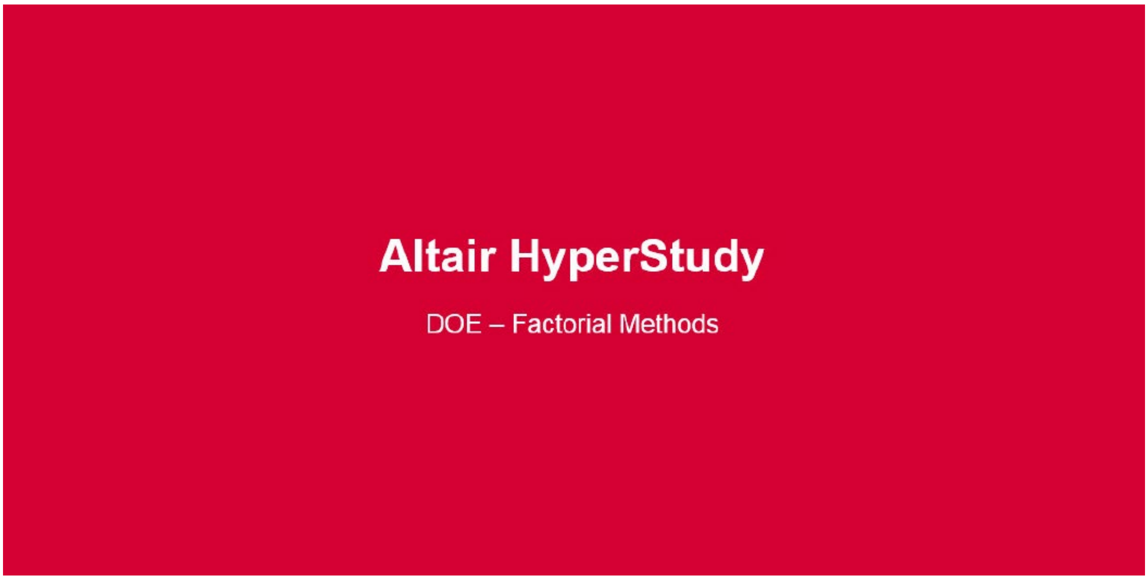
For example, consider eight variables: five variables with two levels, two variables with three levels and one variable with four levels. The number of full factorial runs is 1152 = $2^5 * 3^2 * 4^1$.

Total number of Design Variable	8
Number of design variable with 2 level	5
Number of design variable with 3 level	2
Number of design variable with 4 level	1
Total number of runs	$2^5 * 3^2 * 4^1$

The following picture illustrates the full factorial run matrix for a three-variable problem (variables A and B have two levels and variable C has three levels).

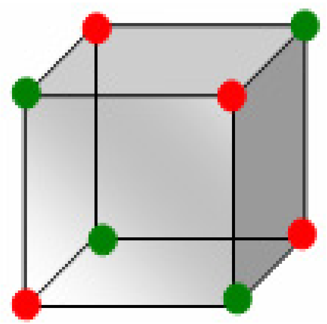
Run Number	Design Variable Name	A	B	C
	Design Variable Level	2	2	3
	Number of Runs	**$2^2 \times 3^1 = 12$**		
1		1	1	1
2		1	1	2
3		1	1	3
4		1	2	1
5		1	2	2
6		1	2	3
7		2	1	1
8		2	1	2
9		2	1	3
10		2	2	1
11		2	2	2
12		2	2	3

5.5.3 Fractional Factorial - Screening DOE



https://altair-2.wistia.com/medias/jv4nw4q0wb

Fractional factorial DOE is a factorial experiment in which only a chosen fraction of the combinations required for the full factorial DOE is run. Fractional factorial designs are used to reduce the number of runs required to extract pertinent information about the main effects and two-factor interactions. The green dots in the image below illustrate the combinations a Fractional factorial design considers.



Three one-third fractions of the 3-factorial design (9 points)

This reduction in computational effort comes at the cost of an inability to completely resolve all of the main effects and interactions. Higher order interactions are often confounded with each other and, in some cases, can be confounded with the two-factor interactions. For example, for factors A, B, and C, each at two levels, only four runs would be required to resolve the main effects. However, all of the two-factor interactions would be confounded with the main effects (A:BC, B:AC, C:AB). Explanation: A:BC may be read as: Main Effect of factor A is confounded/blended with the interaction of B and C.

When applicable, the orthogonal arrays from other schemes, such as Plackett-Burman or Taguchi, are used internally to reduce run count.

The amount of confounding in a Fractional Factorial design is classified by its resolution. Resolution describes the degree of main effect confounding with interactions such as resolution III confounds main effects with 2-level interactions (hence there is no resolution I or II). For our purposes, resolution IV is a good balance between accuracy and effort.

The types of resolutions available in HyperStudy are:

III: The effects are resolved with respect to each other, but all of the effects are confounded with the two-factor interactions.

IV: The effects are resolved with respect to each other and the two-factor interactions, but the two-factor interactions are confounded with respect to each other

V: The effects are resolved with respect to each other and the two-factor interactions, and the two-factor interactions are resolved with respect to each other

Usability Characteristics

- The desired resolution will determine the number of runs in the DOE
- Resolution type III should only be used on applications in which the interactions are known to be small with respect to the effects. This makes the confounding unimportant
- When all variables have only two levels and the resolution is type III, avoiding the confounding between the effects and specified two-factor interactions can be achieved by using the Interactions tab

- The techniques used to generate the run matrix work most effectively when variables have the same number of levels
- In HyperStudy, any data in the inclusion matrix is combined with the run data for post-processing. Any run matrix point which is already part of the inclusion data will not be rerun

Settings

In the Specifications step, you can change the following settings of Fractional Factorial from the Settings tab.

Parameters	Default	Range	Description
Resolution	III	III, IV, V	Select the resolution of the DOE
Number of runs (npt)	Depends upon the selected resolution	Any positive integer	
Use inclusion matrix	False	True or False	Concatenation with duplication between the inclusion and the generated run matrix

5.5.4 Plackett-Burman - Screening DOE

When a large number of factors needs to be screened to identify the ones that may be important (i.e., those that are related to the system performance), we need to employ a design that allows us to test the largest number of factor main effects with the least number of observations.

Plackett and Burman developed highly fractionalized designs to screen the maximum number of (main) effects in the least number of experimental runs in case of two-level factors. Plackett-Burman designs are very economical designs (with the run number a multiple of four rather than a power of two) and are efficient in screening when only main effects are of interest. This is because the main effects in a Plackett-Burman design are, in general, heavily confounded with two-factor interactions. The Plackett-Burman design in 12 runs, for example, may be used for an experiment containing up to 11 factors. PB can only be run for two-level designs and in HyperStudy it cannot be used with more than 13 design variables.

Some important characteristics:

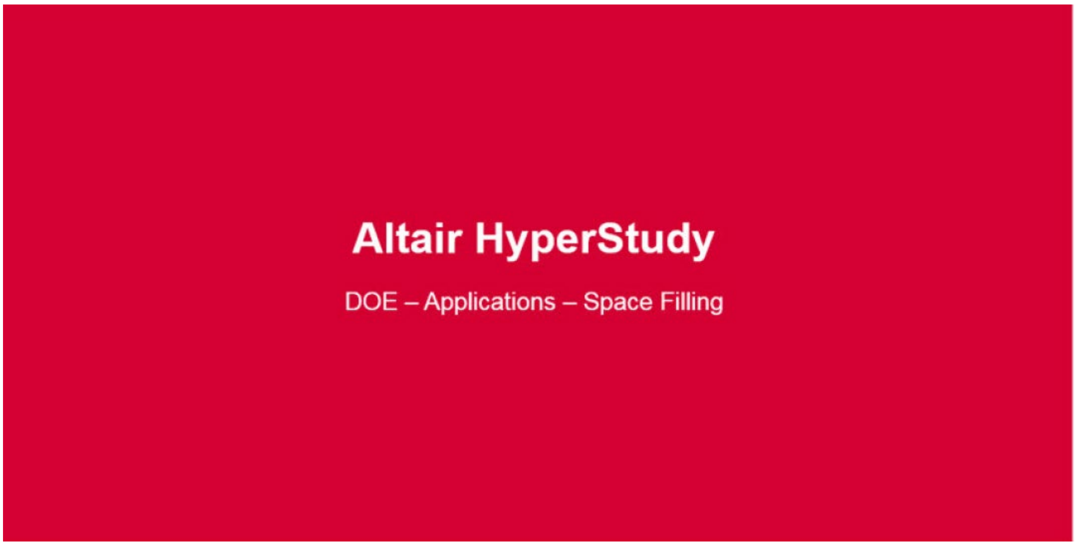
- Only main effects, no interactions
- Only two-level variables can be used
- Two level saturated fractional factorial designs
- Number of runs (N) is always a multiple of 4
- Are used for studies involving up to (N-1) design variables. The maximum number of variables you can use is 35

- Best suited for problems where the interactions are expected to be minimal and the main effects dominate

Run	A	B	C	D	E	F	G	H	I	J	K
1	+	-	+	-	-	-	+	+	+	-	+
2	+	+	-	+	-	-	-	+	+	+	-
3	-	+	+	-	+	-	-	-	+	+	+
4	+	-	+	+	-	+	-	-	-	+	+
5	+	+	-	+	+	-	+	-	-	-	+
6	+	+	+	-	+	+	-	+	-	-	-
7	-	+	+	+	-	+	+	-	+	-	-
8	-	-	+	+	+	-	+	+	-	+	-
9	-	-	-	+	+	+	-	+	+	-	+
10	+	-	-	-	+	+	+	-	+	+	-
11	-	+	-	-	-	+	+	+	-	+	+
12	-	-	-	-	-	-	-	-	-	-	-

Example DOE with 12 variables

5.5.5 DOE for Space Filling - DOE for Response Surface Evaluation



https://altair-2.wistia.com/medias/prfocgu86p

The objective of DOE studies for response surface evaluation is to uniformly distribute the design points in the design space in order to feed into a fit method for accurate prediction of the the model behavior. The studies are performed with only a few factors to obtain higher order response surfaces. This fit can then be used in optimization, stochastic studies and to do quick what-if analysis.

In CAE, the models are already (and becoming even more) demanding, especially if one looks at computer run times. Crash models and especially CFD model runs may last several hours, maybe even days.

Types: Box-Behnken

Central composites

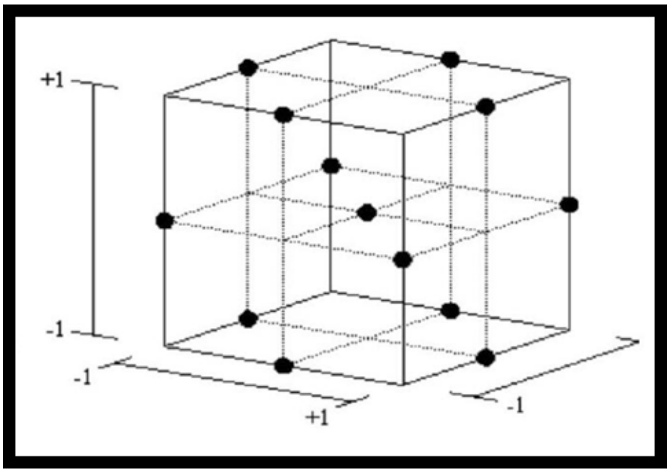
Latin Hypercube

Hammersley

Method of Extensible Lattice Sequences

5.5.6 Box-Behnken - Space Filling DOE

Box-Behnken designs are used to generate higher order response surfaces using fewer required runs than a normal factorial. Box-Behnken designs place points on the midpoints of the edges of the cubical design region, as well as points at the center.



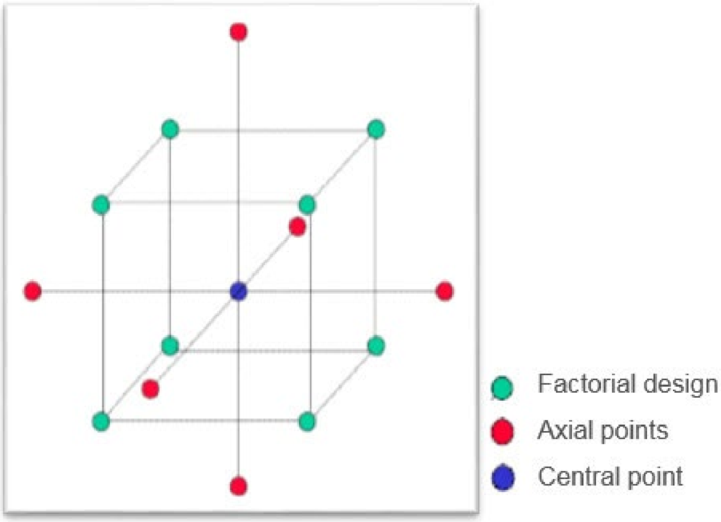
A Box Behnken Design for three factors (13 points)

Some characteristics:

- Box-Behnken designs place points on the midpoints of the edges of the cubical design region, as well as points at the center
- The Box-Behnken design is nearly the opposite of a central composite plan in that it uses the twelve middle edge nodes and three center nodes to fit a 2nd order equation
- Box Behnken allows a second order surface to be generated through the points but requires less points than Central Composite Design (see below)
- Should not be used when accurate predictions at the extremes are important
- Only available when all design variables have 3 levels
- It doesn't explore the corners of the design space which may be considered an disadvantage of Box Behnken

5.5.7 Central Composite Designs - Space Filling DOE

The Box-Behnken design and the Central Composite Design can be visualized as near compliments of each other. They both essentially suppress selected runs from a full factorial matrix in an attempt to maintain the higher order surface definition. For example, for 3 three-level variables, the full factorial run size is 27 (=33). The central composite plan drops all of the middle edge nodes, resulting in only fifteen runs. The Box-Behnken design is nearly the opposite in that it uses the twelve middle edge nodes and the center node to fit a 2nd order equation. Central Composite plus Box-Behnken becomes a full factorial with extra samples taken at the center.



Central Composite Design for 3 design variables at 5 levels (15 points)

Preset name	Axial distance	No of center runs
Rotatable	2	User defined
Orthogonal	1.41421	User defined
Rotatable & Orthogonal	2	12
On Face	1	User defined
User Defined	User defined	User defined

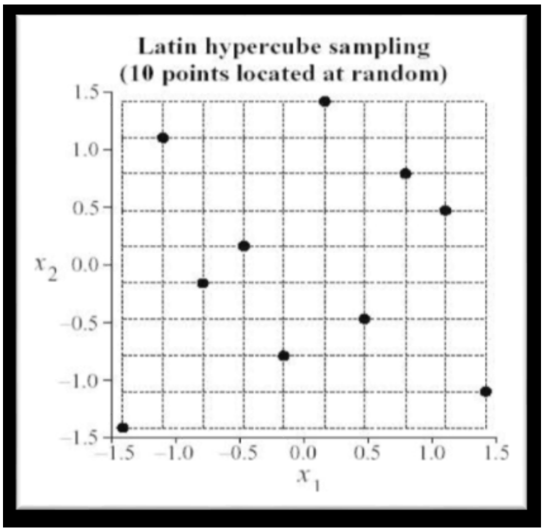
Some characteristics:

- For building a second order (quadratic) model without needing to use a complete three level factorial experiment.
- Parameters:
- Number of center runs
- Axial distance of star pts from center
- Total runs = 2k + 2k + N center points (k = design variables)
- HyperStudy has a limit of 20 design variables when using CCD

From a user experience (personal view of course) it is not used often, also because it is rather tricky to find out (or define) the axial points outside the design space

5.5.8 Latin Hypercube - Space Filling DOE

A square grid containing sample positions is a Latin square if, and only if, there is only one sample in each row and each column.



Latin Hypercube sampling 10 points

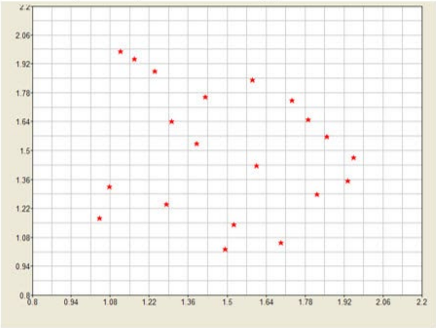
Some characteristics:

- Use case: If you can afford for instance to run 10 designs, then each design variable is split into 10 levels. Then the method picks a combination of design variables in such a way that no one value is used twice. This situation (or call it selection is depicted in the image). Of course, the more points we have, the better populated the design space will be.
- Default=Minimum runs for an approximation = 1.1x ((N+1)*(N+2))/2, where N = number of factors (don't go below this default number of runs)
- Recommended number of runs for most approximations = 1,1(N+1)*(N+2)
- Each design variable is divided into same number of levels (which is also equal to the number of runs)
- Each design variable value is used once
- Latin Hypercube may not sample the limits of the design space

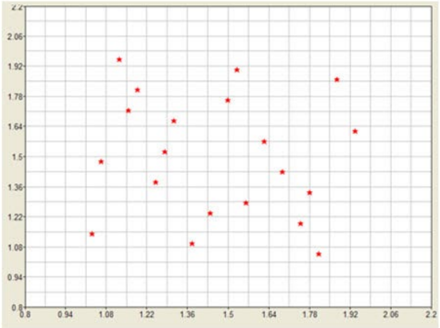
Latin Hypercube design is especially useful in exploring the entire design space and creating fitting functions to the exact responses. To get a good quality fitting function using an LH DOE, a minimum number of runs should be evaluated. (N+1)*(N+2) runs are needed to fit a second order polynomial. Most responses are close to a second order polynomial within the commonly used design variable ranges of -+10%. As a result, we recommend this equation to calculate the number of runs needed or a minimum of 1.1 ((N+1)*(N+2))/2 runs where the coefficient 1.1 is to increase the minimum number by 10% for cross-validation.

5.5.9 Hammersley - Space Filling DOE

Hammersley works in similar way to Latin HyperCube. Hammersley uses a quasi-random number generator, based on the Hammersley points, to uniformly sample a unit hypercube.



Latin HyperCube

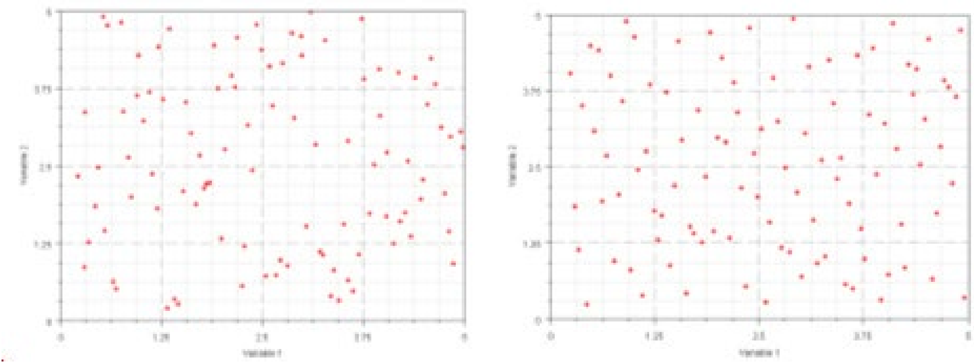


Hammersley

Some characteristics

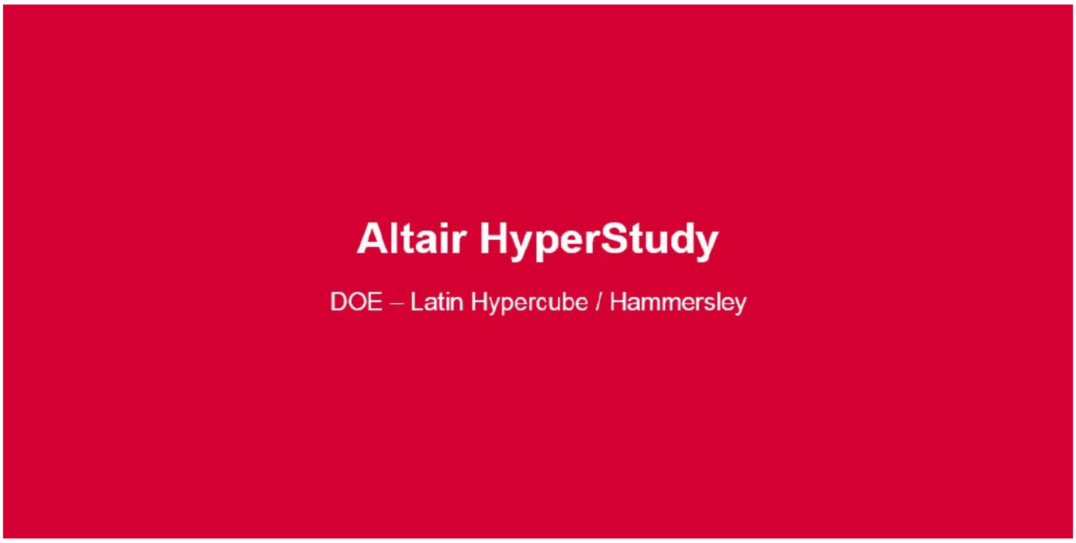
- This technique generates a design similar to the Latin Hypercube but also attempts to distribute the points evenly within the design space
- This provides the best general-purpose DOE type for the generation of an approximation
- Minimum runs for an approximation =1.1 ((N+1)*(N+2))/2, where N = number of factors
- Recommended number of runs for most approximations = 1.1(N+1)*(N+2)

Hammersley has a better coverage of the design space than Latin HyperCube. Look at the image above, in the Latin HyperCube you see more a kind of clustering of design points than in Hammersley; also, there are quite some “white”, i.e. uncovered areas. From a use case perspective, Hammersley is typically used for space filling DOE’s as it has a better design matrix. If the number of allowed runs is increased both Hammersley and Latin HyperCube “converge”



Latin Hypercube (left) and Hammersley (right) for 100 runs

A summary on the last two methods is provided in the following recording



https://altair-2.wistia.com/medias/r9ioggblnn

5.5.10 Modified Extensible Lattice Sequence (MELS) – Space Filling DOE

Note: This method is available with HyperStudy starting version 14.130

A lattice sequence is a quasi-random sequence, or low discrepancy sequence, designed to equally spread out points in a space by minimizing clumps and empty spaces. This DOE type also has the property of extensibility, which means the method can take an existing set of data in a space and add more data points to provide equal coverage. This property makes lattice sequences an excellent space filling DOE scheme. The number of runs is specified by the user.

Usability Characteristics

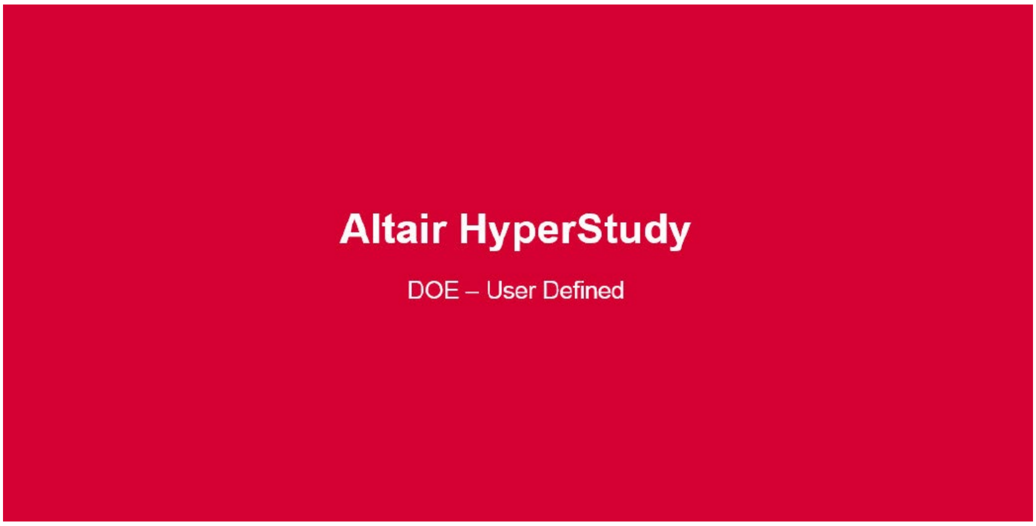
- Use for exploring the entire design space and creating fitting functions to the exact responses. It is the recommended default space filling scheme
- To get a good quality fitting function, a minimum number of runs should be evaluated. (N+1)(N+2)/2 runs are needed to fit a second order polynomial, assuming that most responses are close to a second order polynomial within the commonly used design variable ranges of -+10%. An additional number of runs equal to 10% is recommended to provide redundancy, which results in more reliable post-processing. As a result, this equation is recommend to calculate the number of runs needed or a minimum of 1.1*(N+1)(N+2)/2 runs (in other words, minimum number of suggested runs in space filling DOEs is 1.1 times Hammersley or Latin HyperCube)
- Add existing data to the inclusion matrix to use the extensibility feature. While any data can be used as an inclusion, the best performance can be expected when the inclusion is an existing data set from a MELS DOE

Settings

In the Specifications step, you can change the following settings from the Settings tab (see summary next page).

Parameter	Default	Range	Description
Number of runs	$\frac{1.1(N+1)(N+2)}{2}$	> 0 integer	Number of new designs to be evaluated.
Random Seed	1	Integer 0 to 10000	Controls the repeatability of runs, depending on the way the sequence of random numbers is generated. 0 random (non-repeatable) > 0 triggers a new sequence of pseudo-random numbers, repeatable if the same number is specified.
Use Inclusion	false	true or false	The use of an inclusion matrix will trigger the DOE to be extensible as it tries to fill in the space already covered by the existing data set.

5.5.11 User-Defined Design



https://altair-2.wistia.com/medias/e9joole6gf

The user-defined design allows you to load your own design to perform parametric studies. The user-defined design is read in by HyperStudy and is used like any other design. The advantage of using a user-defined design is that you can create your own design based on individual requirements. You must specify the number of runs (rows) and the number of columns (=variables) in the specified matrix in the first row. Variable values are defined as the level. Spaces, tabs, or commas can delimit the individual elements of the matrix.

The user defined matrix uses integers to represent the corresponding level of the variable. This is in contrast to the run matrix DOE (see further below), which contains exact values of the variables.

An example of a user-defined design is shown below.

9	3	
1	1	1
1	2	2
1	3	3
2	1	3
2	2	2
2	3	1
3	1	2
3	2	3
3	3	1

First row – number of runs / designs; number of variables; 1=level 1, 2= level 2, 3= level 3

5.5.12 Run Matrix

When you are using Run Matrix, you can use your own design matrix to perform parametric studies. The run matrix is read in by HyperStudy and used as is. The advantage of using a run matrix directly is that you can create your own design based on individual requirements. The matrix does not have to fit any DOE type requirements. It can be used to automate a parameter study. The run matrix uses exact values of the variable. This is in contrast to the user defined DOE, which contains integers to repre-sent the corresponding level of the variable. It is not necessary to utilize all designs in a study. Designs that are not desired can be turned off from the Write/Execute runs panel. Spaces, tabs, or commas can delimit the individual elements of the matrix. The rows define the different designs; the columns define the design variables.

An example of a run matrix is shown below:

1.0	2.0	1. 3.0	2. 4.0
4.1	4.3	4.5	4.6
6.7	8.1	10.0	11.0
17.2	1.0	1.0	3.0
.02	0.4	0.5	1.7
3.4	2.1	7.3	9.1

The Run Matrix uses exact values of the variable. Rows=runs; columns=design variables

5.5.13 None

Again, allows to define your own matrix, this time however, you are using Test Data consisting of variables (DV) and responses R1 (=results, measured values)

DV1	DV2	R1
150	52	47
132	98	54
198	42	67

5.5.14 Summary DOE Study Types / Methods

A summary of the above summarized methods is shown in the graphs below (taken from the Online Help).

Method	Parameter Screening	Space Filling	Custom	DV Levels	Continuous	Discrete	Basic Parameters	Properties and Comments	Categorical
Fractional Factorial	✓			2 or 3	✓	✓	Select the appropriate resolution.	Resolution indicates the level of accuracy of the interactions. Interactions should not be used with Resolution III.	✓
Full Factorial	✓			Any	✓	✓		Requires a high number of simulations and is therefore unsuitable for most studies. Total number of runs should be less than 1,000,000.	✓
Plackett Burman	✓			2	✓	✓	You can either click ***Apply*** for **AutoSelect** or select a table using the **Design** pull-down menu.	Computationally least expensive. Number of points can be 12, 20, 24, 28 or 36. Selecting ***Autoselect*** will pick pbdgn12 if $N < 12$, where N is the number of design variables; pbdgn20 if $12 <= N < 20$, etc. Limited to 35 design variables. Categorical variables must have exactly two levels.	✓
Central Composite Design		✓		5	✓			Use this method when the responses are known to be quadratic. Limited to 20 design variables.	
Box-Behnken		✓		3	✓	✓	You can either click ***Apply*** for **AutoSelect** or select a table using the **Design** pull-down menu.	Use this method for building quadratic response surfaces if the responses are known to be quadratic and predictions are not required at the edge of the design space. Number of points can be 13, 25, 41, 49. 57. Selecting ***Autoselect*** will pick bbdgn13 if $N < 4$, where N is the number of design variables; bbdgn25 if $N = 4$, bbdgn41 if $N = 5$, etc. Limited to 7 design variables. Discrete variable must have at least 3 levels. Categorical variables must have exactly 3 levels.	✓
Latin Hypercube		✓		Any	✓	✓	You can either accept the default number of runs or enter a different value.	Use this method when the response surface is highly nonlinear. The default number of runs is $1.1*((N+1)*(N+2))/2$, where N is the number of design variables. You must maintain the value of the random seed in order to get repeatable designs.	✓

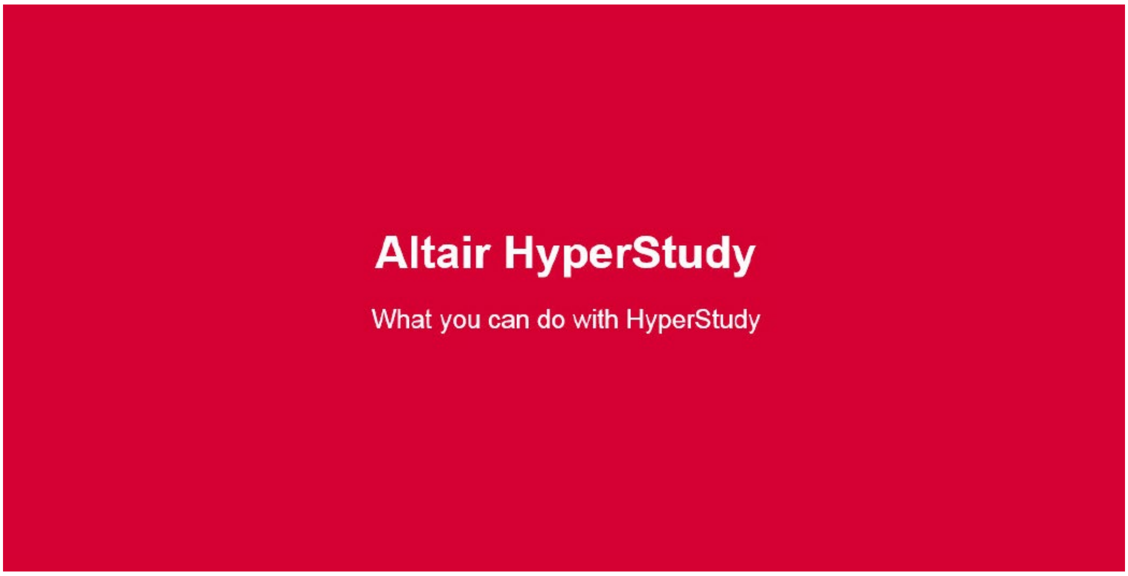
continuation

Method	Parameter Screening	Space Filling	Custom	DV Levels	Continuous	Discrete	Basic Parameters	Properties and Comments	Categorical
Modified Extensible Lattice Sequence (MELS)		✓		Any	✓	✓	You can either accept the default number of runs or enter a different value.	Use this method when the response surface is highly nonlinear. This method is a better space filler than Latin Hypercube. The default number of runs is 1.1*((N+1)*(N+2))/2, where N is the number of design variables.	✓
Hammersley		✓		Any	✓	✓	You can either accept the default number of runs or enter a different value.	Use this method when the response surface is highly nonlinear. This method is a better space filler than Latin Hypercube. The default number of runs is 1.1*((N+1)*(N+2))/2, where N is the number of design variables.	✓
Taguchi	✓			Varies	✓	✓	You can either choose AutoSelect or a specific design matrix.	The levels of each variable must be set accordingly to ensure compability with a specific design matrix.	✓
User-defined Design			✓	Any	✓	✓	Select the perturb file.	Use this method to create a design matrix using abstract variable levels.	✓
Run Matrix			✓	Any	✓	✓	Select the perturb file.	Use this method to create a design matrix using literal variable values.	
None									

6 Introduction to HyperStudy

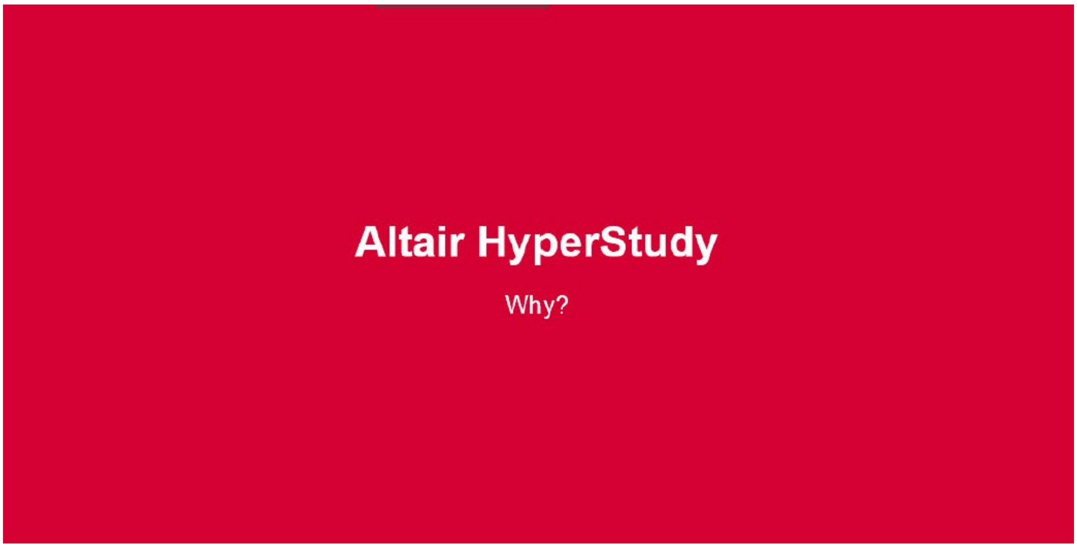
HyperStudy enables you to explore, understand and improve your system's designs using methods such as design-of-experiments and optimization. HyperStudy generates intelligent variations of the parameters of any system model and reveals relationships between these parameters and the system responses. By using HyperStudy, you can make better decisions and optimize the performance, reliability and robustness of your systems.

In other words, with direct connections to Altair's Solvers, interfaces to several other solvers, and the ability to interface with any solver, HyperStudy is the tool when it comes to DOE, Optimization, and Robust Design.



https://altair-2.wistia.com/medias/hadvekycdk

The design of HyperStudy as a wizard makes it very easy to learn and use. It is applicable to study the different aspects of a design under various conditions, including non-linear behaviors and multi-physics environments. It can also be used for multi-disciplinary optimization of a design combining different analysis types.



https://altair-2.wistia.com/medias/a7oa6ekru7

If you have set up a model in HyperMesh, for example, you can invoke HyperStudy which then has direct access to model variables. If you're using a solver that has a text input file that cannot be read to HyperMesh, you can parametrize it using the HyperStudy editor.

HyperStudy uses its interfacing abilities both to generate additional input files for the solver (with varied design) for each experimental measurement, and to read in evaluated responses.

Finally, if you're working with a solver that has no existing interface to HyperMesh or HyperGraph, you could use the Templex programming language to build a custom interface.

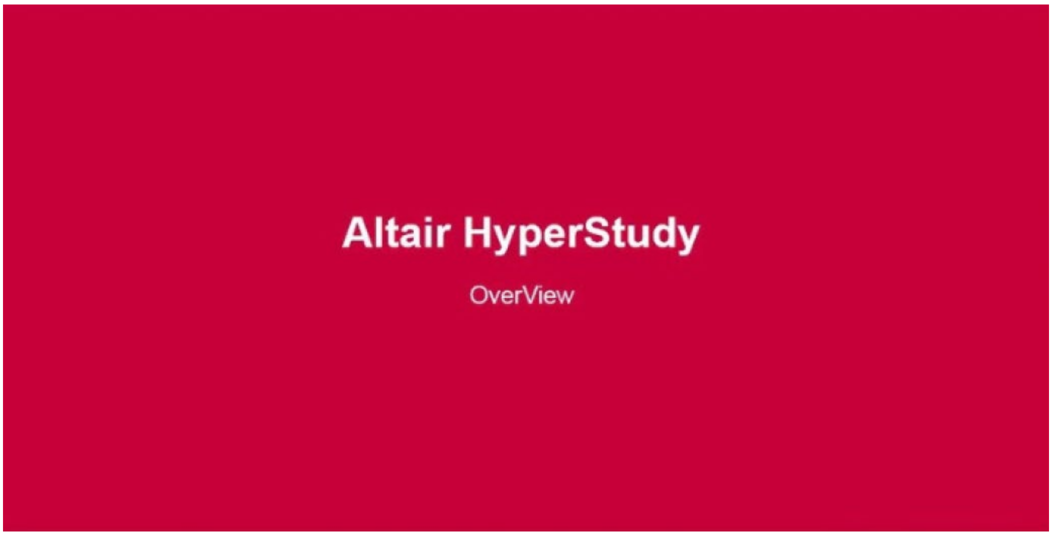
The models can be parameterized very easily. Besides the typical definition of solver input data as design variables, the shape of a finite element model can also be parameterized with ease through HyperMorph; morphing technology in HyperMesh.

HyperStudy Post-Processing module contains display, analysis and data mining capabilities that helps engineers to overcome the challenging task of extracting relevant information from multi-run studies. With its unique and powerful suite of tools, simulation results can be analyzed, sorted and studied effectively in HyperStudy.

Specifically developed for design of experiments (DOE), stochastic simulations, and optimization techniques, HyperStudy users can:

- Gain insight into the physics of a design
- Assess the robustness of a design for controlled or uncontrolled variations in the design parameters
- Optimize a design for multi-disciplinary attributes

Our colleague Ken Mix summarized some key aspects of and about HyperStudy in the following video recording.



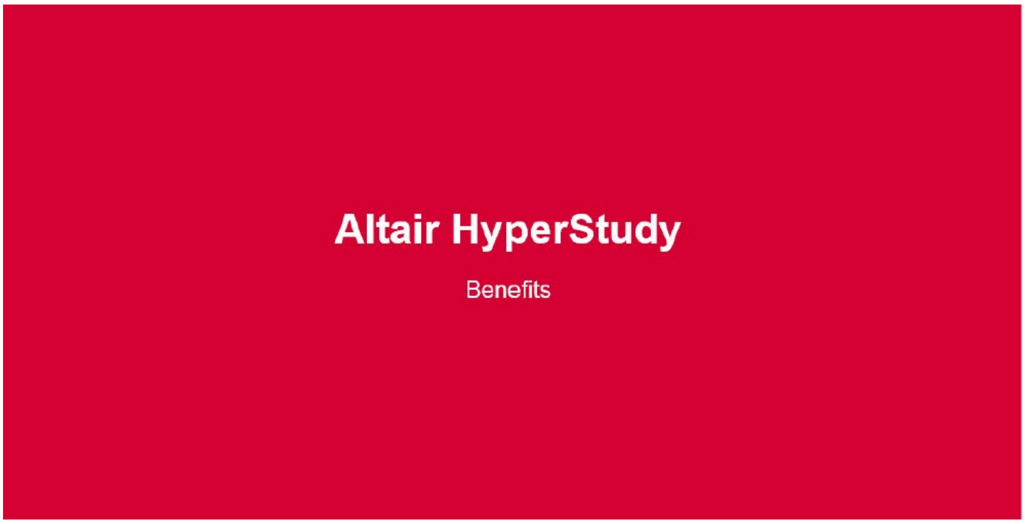
https://altair-2.wistia.com/medias/d0m4p4xhie

6.1 HyperStudy Benefits

1. Provides engineers with an easy way to study effects of design changes for complex analysis events.
2. Allows engineers to assess the robustness of designs and provides the guidance necessary to achieve robust designs.

3. Allows engineers to perform multi-disciplinary optimization studies for different attributes of a design.
4. Allows engineers to perform system identification and correlation studies of designs;
5. Complements existing CAE software with added functionality and direct interfaces to major solvers.
6. Minimizes time-to-market by identifying design direction for difficult problems

The HyperStudy benefits are summarized by Fatma Koçer (Altair) in the next video



https://altair-2.wistia.com/medias/041rhhu2a7

6.2 Before the Study

Remember that one of the main reasons we are using HyperStudy is that CAE is quite computationally intensive. It is prudent, therefore, to spend some time planning your attack strategy. Without adequate planning, it is easy to find you have invested more time and effort than you had intended to.

As with all experiments: Too much data is a reduced return on investment (more effort but no new information). In contrast, the negative side from having too little data is poor results (which makes the efforts wasted). The effort involved in collating and interpreting results is all too often underestimated. To make things easier for you, HyperStudy follows a “wizard” approach. That is, the interface provides the various functions as a step-by-step sequence, ensuring that you complete the steps in the correct order.

Even before you start building your model, you should be clear on the answers to the following:

- What are the mathematical properties of the design variables? Are they continuous, discrete, discretized, integers, etc.? These will take a role in determining which methods are applicable or more effective. Are the objectives and constraints suitable? Are they physically meaningful? What are their mathematical properties?
- Can the responses be measured quantitatively? What accuracy is required for a meaningful experiment?
- Are there any effects that should be deliberately blocked or omitted? Should a screening-run be performed to verify assumptions on importance? What levels should be used for the screening run?

- Which effects will be aliased or confounded by the screening run? Are they important?
- What should the sequence of experiments be? Can the results of the first experiment be used to create the second, and so on?
- How will findings be confirmed?
- Last but not least did you line up your resources for these experiments such as hardware, software and time?

Also spend some time planning how you will collate and present results. FE models are often interpreted by displaying results on the 3D model of the component, but this may not be appropriate. Remember that you will have to review a much larger magnitude of results: you may have results of several dozen FE analyses!

How you use HyperStudy depends on your model type, your simulation software and your design objectives among other factors.

You can use HyperStudy as a standalone software or you can start it from another HyperWorks product, like HyperMesh Desktop.

Below are some common use cases for setting up a study:

Study Question or Scenario	Best HyperStudy Options
I have a RADIOSS model that I want to do a size/ shape optimization study with. I will run the simulations in my PC	Once you are finished creating your shape variables in HyperMorph, start HyperStudy from HyperMesh' Applications menu. Your model type is "HyperMesh".
I have a MotionSolve model. I will run the simulations in my PC.	Start HyperStudy from MotionView 's Applications menu. Your model type is "MotionView".
I am using a commercial solver that is not in HyperWorks, but it integrates with HyperWorks (Abaqus, LS-DYNA, Design Life, etc.). I would like to set up a size optimization study and will run the simulations in a cluster.	Start HyperStudy in standalone mode. Your model type is "Parameterized File". You can create the parameterized file by using the Editor option under the Tools menu. Register the solver by using the Register Solver Script option in the Edit menu.
I did my analysis in a spreadsheet. Some of the cells are design variables and some are responses.	Start HyperStudy in standalone mode. Your model type is "Spreadsheet".
I calculated my responses using analytical equations.	Start HyperStudy in standalone mode. Your model type is "Internal Math"

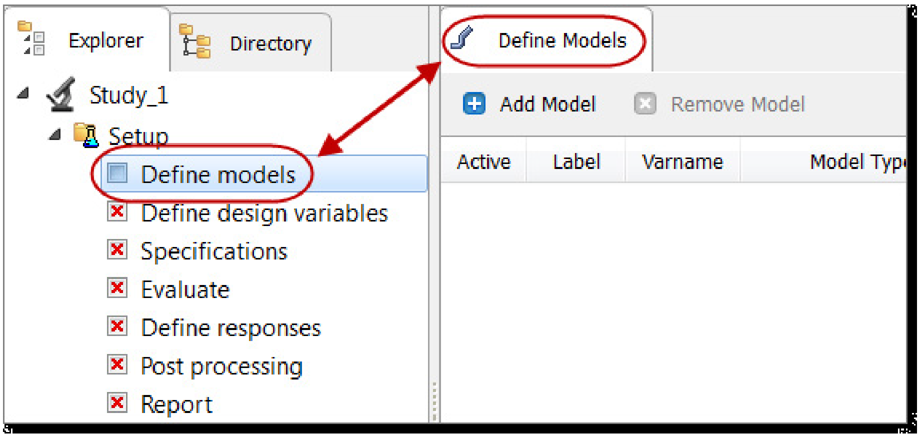
Once you have finished setting up your study, you will need to select one or more study approaches to find the answers to your study questions. The best combination of approaches and the best method to use for each approach depends on your application and objectives.

Below are examples of approach choices for specific scenarios:

Study Approach Question or Scenario	Best HyperStudy Options
Which design variables have a significant effect on my responses?	Use a parameter screening DOE, such as Fractional Factorial. Once the parameter screening DOE is complete, look at the Effects and Interactions plots.
How can I do quick trade-off studies?	Use a space-filling DOE, such as MELS to create fitting functions to your responses.
What are the best design variable values to minimize? my objective, while meeting my design requirements? What is the reliability of my design?	Use a Stochastic approach and add your reliability assessment.

6.3 Performing the Study

In HyperStudy, a study is a self-contained project (saved into a study file, with a .xml extension) in which models, variables, responses, and approaches are defined. It's a good idea to create a separate folder for each study.



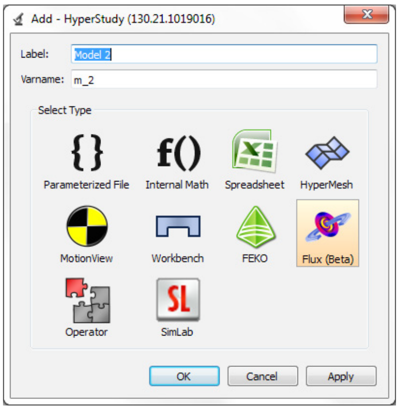
The HyperStudy GUI was designed to guide you through a series of steps to set up a study and/or to add an approach. Each step must be completed before progressing on to the next step.

When you create a study or open an existing study, the study opens in the "Explorer", and any files associated with the study will be displayed in the "Directory". Specify the directory location when you create a study. When opening an existing file, the directory location defaults to the location of the file selected.

As you navigate through the Explorer (by clicking on the steps), related information becomes available in the work area for development or analysis (as with "Define models" in the image below). Often, several tabs will be displayed in the work area.

The Study Setup, contains the following working steps:

1. Definition of the Model Type



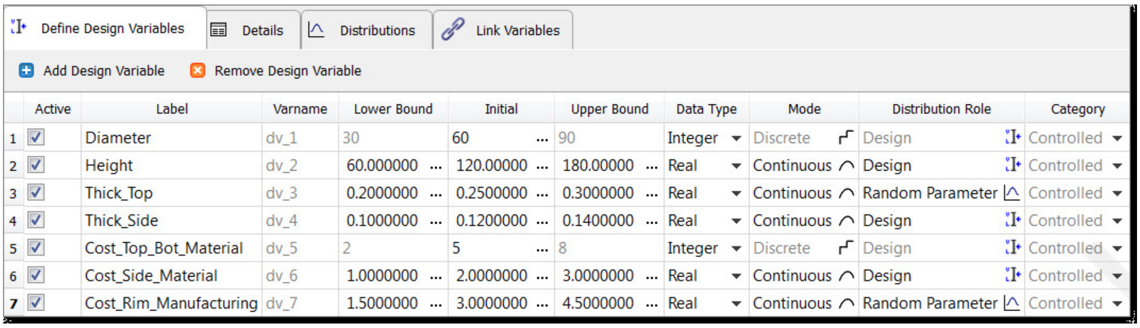
- Parameterized File: Select a Parameterized File model if you want to use a template file. A HyperStudy template file is an ASCII text file that contains the parameterized version of the solver input file (e.g. OptiStruct, RADIOSS, AcuSolve, etc.). The template file extension is .tpl. The most basic HyperStudy template file is a data file in which certain data fields are replaced with parameters using a Templex statements (very important; see example further below)
- Internal Math: Select an Internal Math model if you want to use the HyperStudy response Expression Builder to calculate responses as functions of design variables or responses. As shown n the example further below, Internal Match is also used to process test data (e.g. Excel sheet with variables/factors and results/responses)
- Spreadsheet: Select a Spreadsheet model if your calculations are done in an Excel spreadsheet (in other words, some math is done inside the Excel sheet). With this model type, you must load the spreadsheet you want to work on in the Resource column. Once you load a .xls file, HyperStudy automatically defines the Solver input file and Solver execution scripts columns (see example below).
- HyperMesh: HyperStudy is started from within HyperMesh; HyperStudy then has direct access to the HyperMesh database of the model to perform easy design parameterization
- The following user profiles and solvers are supported:
- Abaqus, ANSYS, LS-DYNA, Nastran, OptiStruct, PAM-CRASH 2G, RADIOSS
- MotionView: HyperStudy is started from within MotionView; HyperStudy then has direct access to the MotionView database. The following MotionView solvers are supported:
- MotionView, Adams, DADS, Abaqus, Nastran
- Workbench: Select a Workbench project file to import design variables and responses from ANSYS Workbench. HyperStudy recognizes ANSYS Workbench Input parameters as design variables and Direct Output parameters as responses
- SimLab: Select a SimLab project (.sls) file to import design variables from SimLab
- FEKO: Once a FEKO (.cfx) file has been added to HyperStudy, models parameters are automatically identified
- Flux: Once the Flux link file (that has been generated previously from Flux) has been added to HyperStudy, models parameters and output functions are automatically identified and imported

2. Definition of Design Variables, or Factors

System parameters that you can change to improve the system performance. Examples of design variables are beam dimensions, material properties, diameter, and number of bolts. Design Variables can be continuous or discrete. A design variable can be a numerical value, as is common with numerical analyses. Since HyperStudy is a general-purpose tool, the design variable can be quite general - it can even be a text-string. Remember that the levels of a design variable are meaningful only if it is discrete. If you want to carry out a screening run with continuous design variables, you can define them as discrete for the purpose of the screening. If the variable is continuous, you specify the bounds for each, instead of the levels. The subsequent DOE will decide which levels to use within these bounds, depending on the type of design specified for the experiment.

In the Define design variables step, there are four tabs: Define Design Variables, Details, Distributions and Link Variables. In all four tabs, you can add and/or remove design variables to your model. In the first three tabs you can review and edit the following design variable properties:

- Active/Inactive status (Once a variable is unselected; that is. made inactive; it is no longer a variable but is fixed to its initial value)
- Bounds and initial value
- Data type between real and integer
- Mode between continuous, discrete, or categorical



	Active	Label	Varname	Lower Bound	Initial	Upper Bound	Data Type	Mode	Distribution Role	Category
1	☑	Diameter	dv_1	30	60	90	Integer	Discrete	Design	Controlled
2	☑	Height	dv_2	60.000000	120.00000	180.00000	Real	Continuous	Design	Controlled
3	☑	Thick_Top	dv_3	0.2000000	0.2500000	0.3000000	Real	Continuous	Random Parameter	Controlled
4	☑	Thick_Side	dv_4	0.1000000	0.1200000	0.1400000	Real	Continuous	Design	Controlled
5	☑	Cost_Top_Bot_Material	dv_5	2	5	8	Integer	Discrete	Design	Controlled
6	☑	Cost_Side_Material	dv_6	1.0000000	2.0000000	3.0000000	Real	Continuous	Design	Controlled
7	☑	Cost_Rim_Manufacturing	dv_7	1.5000000	3.0000000	4.5000000	Real	Continuous	Random Parameter	Controlled

In the “Link Variables” tab, you can link design variables to each other to reduce the number of independent variables and/or of different models in multi-model studies to have synchronized design updates between models. You can also link the design variable of a model to the responses of other models. If the input of a model is a function of outputs of other models, you can add a design variable for the input of the dependent model and responses for the outputs of the independent models and link the design variable to the responses by entering the dependency in the expression field.

3. Specifications

In the Specification step, the following task(s) may be conducted:

Nominal Run

Runs one simulation where the design variable’s values are set to the initial values.

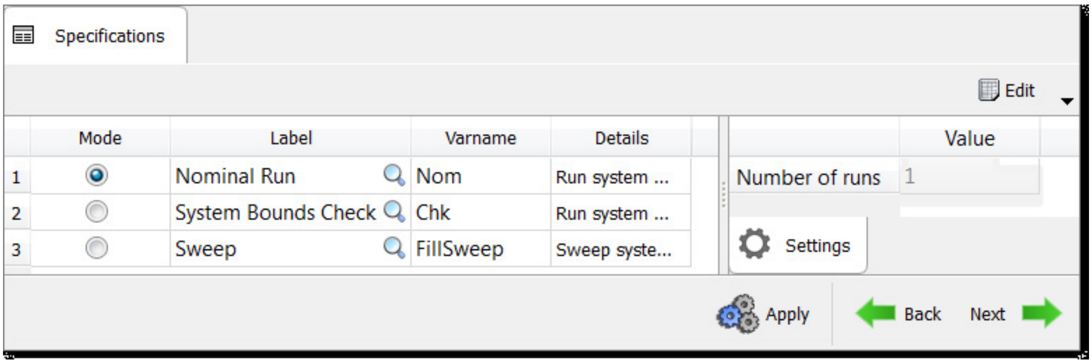
System Bound Check

Checks the study setup and the design space using three runs. The first run sets all of the design variables to their nominal values, the second run sets all of the values to their lower bounds, and the third run sets all of the values to their upper bounds.

Sweep

Evaluates design alternatives for a study where the design variable values increase in equal increments. For example, the first run sets the design variables' values to their lower bounds, the last run sets the design variables' values to their upper bounds, and the runs in between will be set at equispaced incremental values that are determined by the number of runs you specified. By default, five simulations are running.

In sweep study, the values for a discrete variable are iterated by index. If the number of runs exceeds the maximum index of the discrete list, the indexing scheme returns to the first index in a periodic sense.



Definition of responses. These are chosen from the results of the evaluation carried out by the high-fidelity model. For instance, weight, volume, displacement, strain, reaction forces, frequency or stress (i.e. solver dependent you either reference a text-output file or a binary results file).

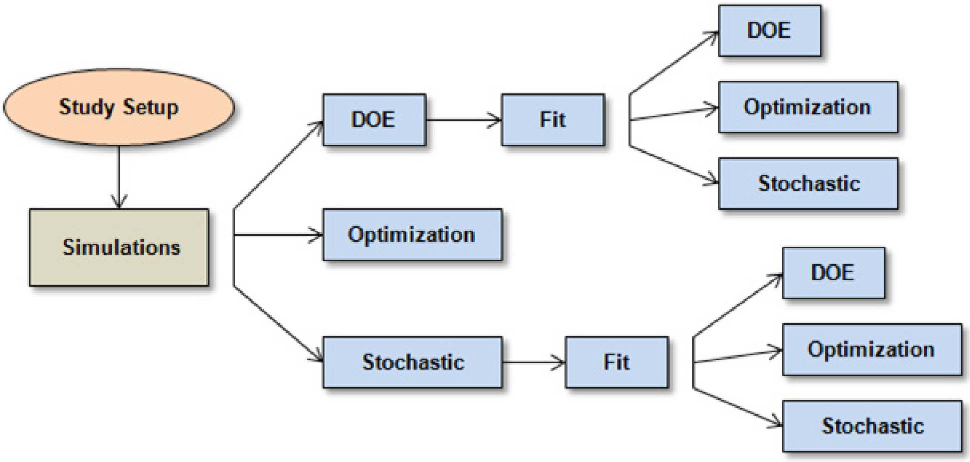
Finally, the

4. Approach (Study Approach) is determined:

In rather general words: A study approach is a specific set of steps taken to study the mathematical model of a design. In HyperStudy, there are four different approaches:

- DOE
- Fit (In cases where simulation resources are scarce, the objective is to have a quick trade-off tool and/or noisy responses need to be smoothened to increase the effectiveness of further studies; you need to consider using fitting functions instead of the exact responses from simulations)
- Optimization
- Stochastic

Each approach serves a different purpose in the design study. The required steps for each approach is similar but there are small differences when needed. For example, you can use the DOE approach (focus of this eBook) if you need to learn the main factors affecting your design, but you need to use the optimization approach if you want to find the design that achieves the design objectives while satisfying design requirements.



However, we focus on DOE Studies in this book:

Once the study has been defined (Study setup completed), you specify what type of experiment you want to use.

Remember that:

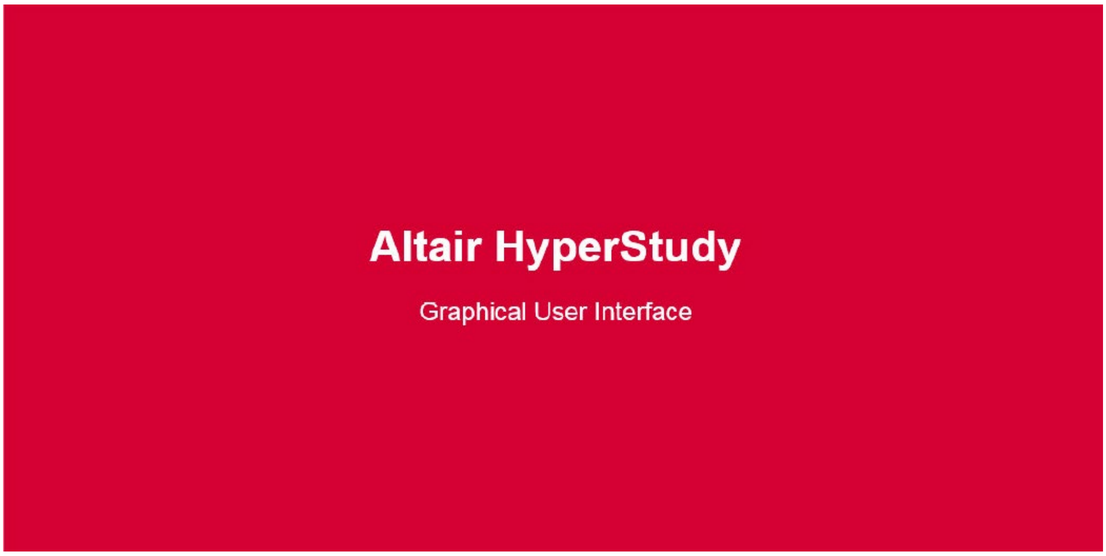
- Full Factorial is not recommended if # factors > 5, since the combination of the number of factors and their levels can make it prohibitively expensive
- Fractional Factorial is often used with just 2 levels for each factor
- Modified Extensible Lattice Sequence (MELS) is our recommended (default) space filling scheme

For the study, you will need to distinguish between controlled and uncontrolled variables. The former consist of design variables that you want to manipulate as a part of your design, while the latter are due to uncontrollable noise. You can use different DOE's for the controlled and uncontrolled variables, depending on the amount of effort you can afford to spend.

Once you have done this, you fine-tune your experiment. Recall the table drawn up earlier when the Fractional Factorial design was discussed. Most DOE's are specified as a matrix, showing the levels that will be used for each factor, and which effects will be confounded. Fine-tuning means you can edit the matrix to change the allocations for each factor / level and the interactions between them.

Once the DOE has been specified, HyperStudy runs the analyses and extracts responses for each of the designs.

6.4 Invoking HyperStudy



https://altair-2.wistia.com/medias/ufdvcil19t

You can invoke HyperStudy as a standalone product, run it in batch mode, or from within HyperMesh and MotionView using the "Applications" menu. The interactive standalone mode can also be accessed through HyperGraph and HyperView using the Applications menu.

HyperStudy Start Options

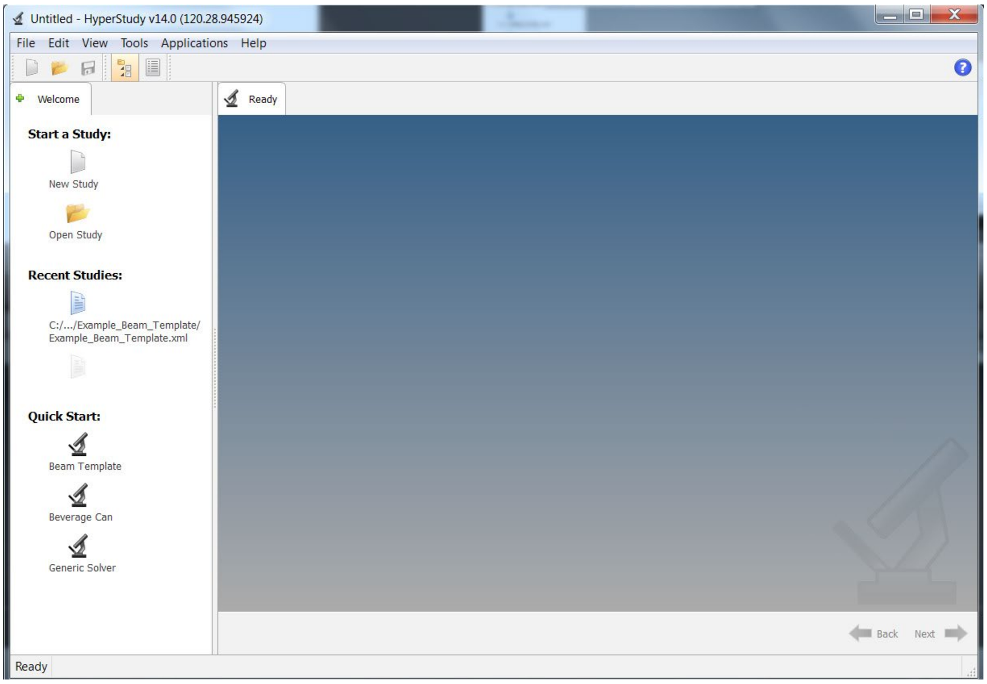
There a number of useful options that you can use to start HyperStudy in UI or in batch. Please have a look at the Online Help documentation for more details.

Also, to be mentioned is the information available in the Online Help Study Contents → Setup-Define Models → Specify Solver Input Arguments

A Solver input argument is any argument to be passed to the solver. The default argument is $file, which means that the qualified solver input file name is passed to the solver script.

- For **LS-DYNA**, this field might contain: `i=$file MEMORY=5000000`
- For **NASTRAN**, this field might contain: `Batch=no`
- For **MADYMO**, this field might contain: `-fg <filename>.xml`
- For solver scripts running on Linux, this field might contain: `-nobg`
- For **Adams**, this field might contain:
- For **Abaqus**, this field might contain: `job=<filename>.inp memory=200Mb interactive`

Once HyperStudy is started you will see the following Graphical User Interface



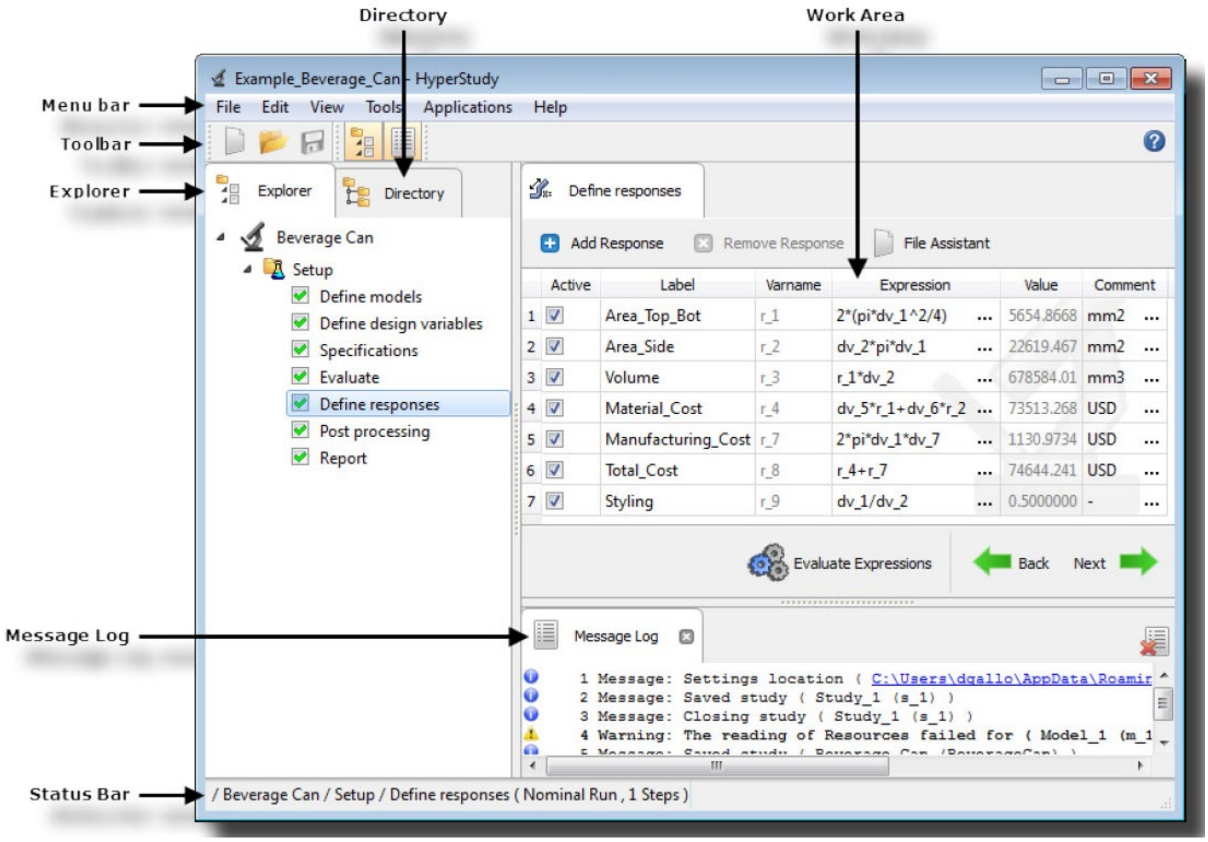
The “HyperStudy Welcome” page as shown above gives you easy and direct access to “Start a Study”, or to access a “Recent Study” (here only one recent study is listed).

Note: To open a recent study (up to 6 studies are listed) just click on the study of interest will open it.

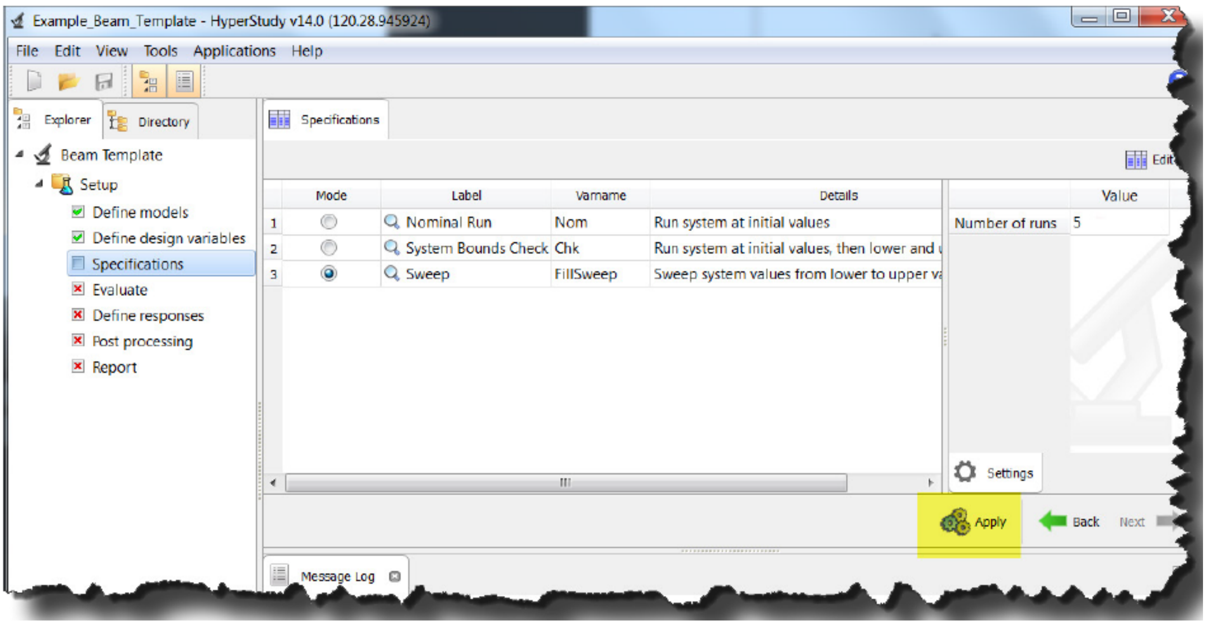
Also note the “Quick Start” in the lower left area of the browser which allows to start the example by clicking on it (Beam Template =OptiStruct Example; Beverage Can = Internal Math Example; and an example regarding a Generic Solver). The studies are set-up already, hence allowing to learn more about the set-up or even to demo it.

6.5 Graphical User Interface

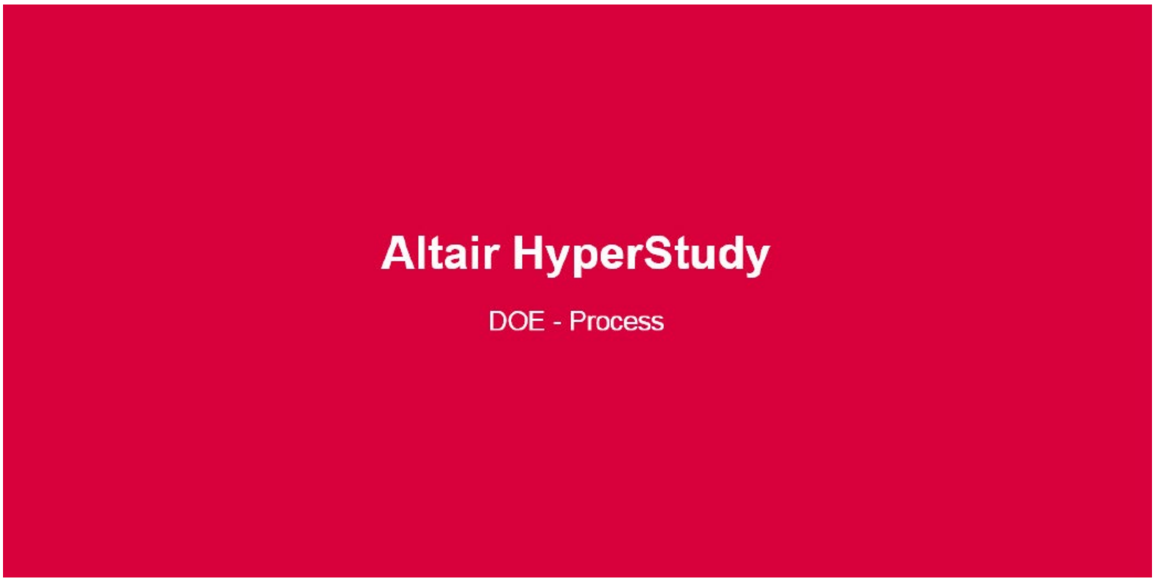
The Graphical User Interface (GUI) was strategically designed to guide you fluently through the study setup and investigation process. Use the GUI to take full advantage of the HyperStudy capabilities needed to quickly learn about the system you are studying (the GUI is depicted on the following page).



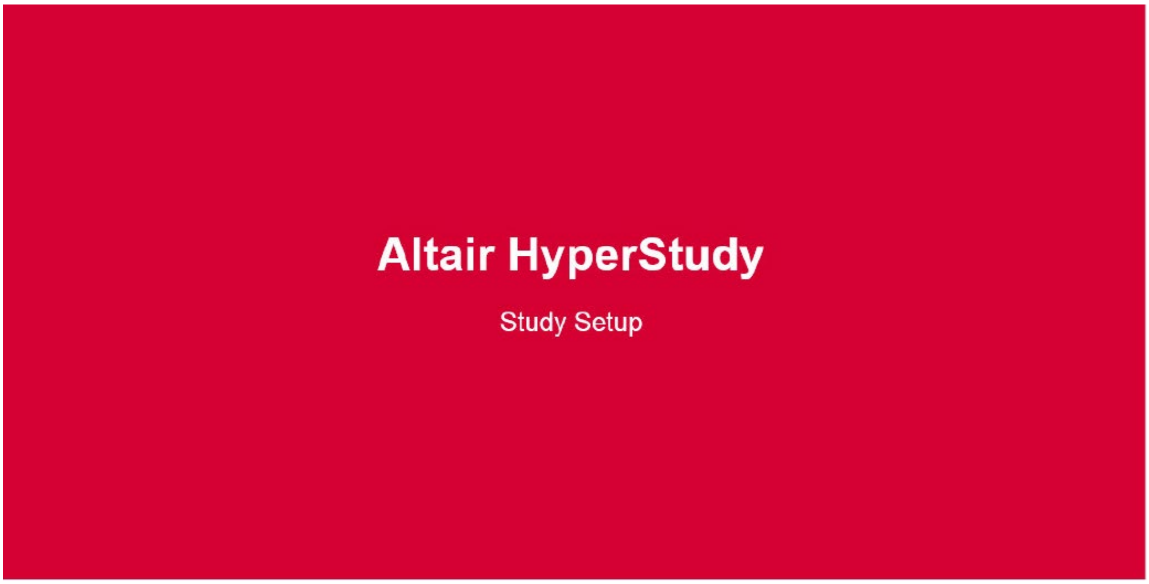
Once you started your study-set up and you make changes e.g. inside the Specifications as shown in the image below, the Apply button (highlighted in yellow) needs to be activated to confirm the changes/settings. If your settings are consistent, the system "allows" you proceed to the next step.



6.6 How to use HyperStudy

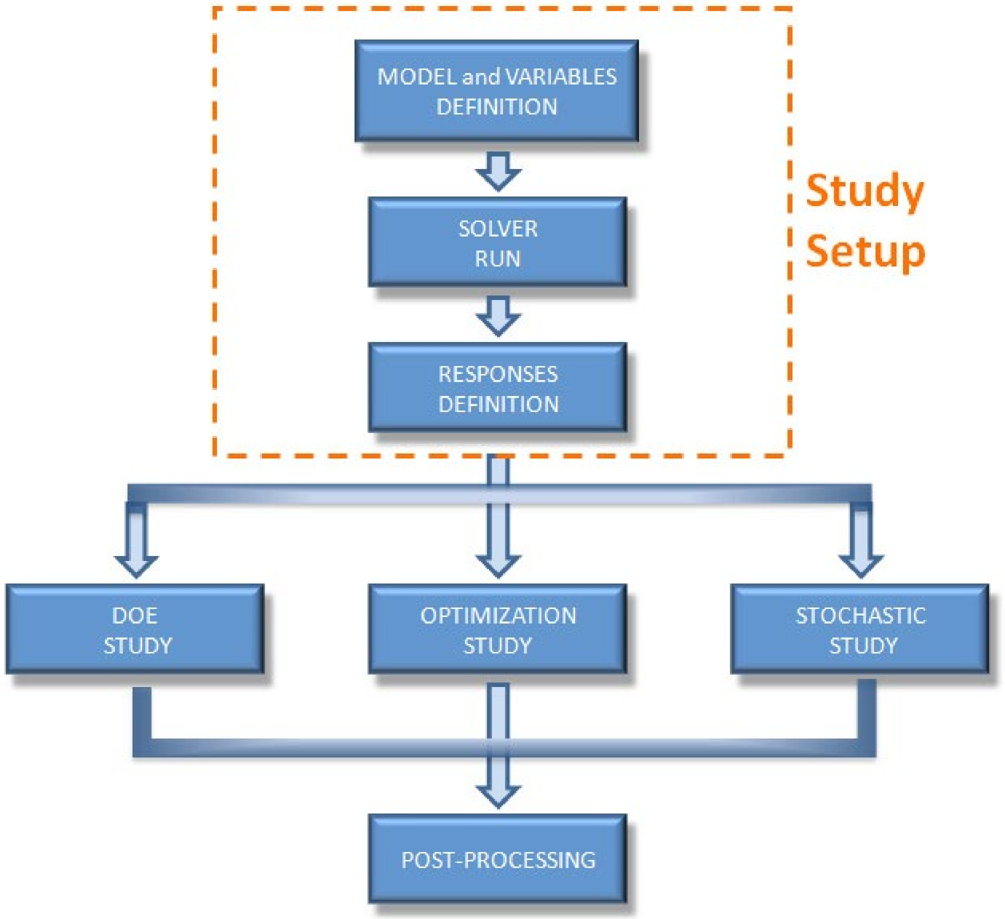


https://altair-2.wistia.com/medias/kelhte75ms

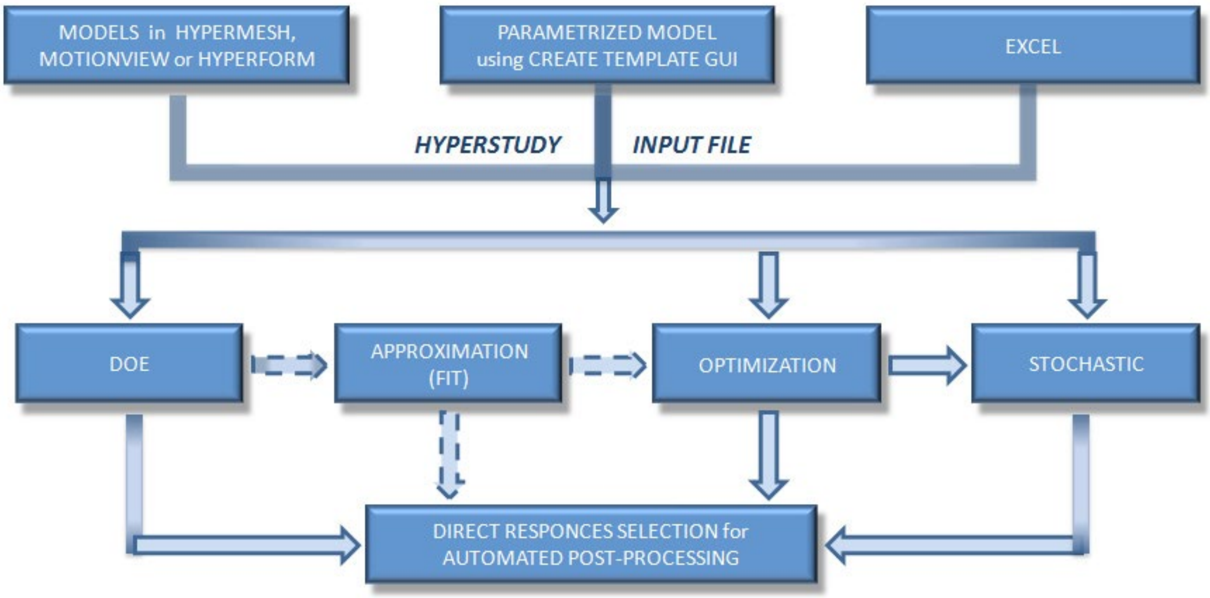


https://altair-2.wistia.com/medias/cbhvf2tu2j

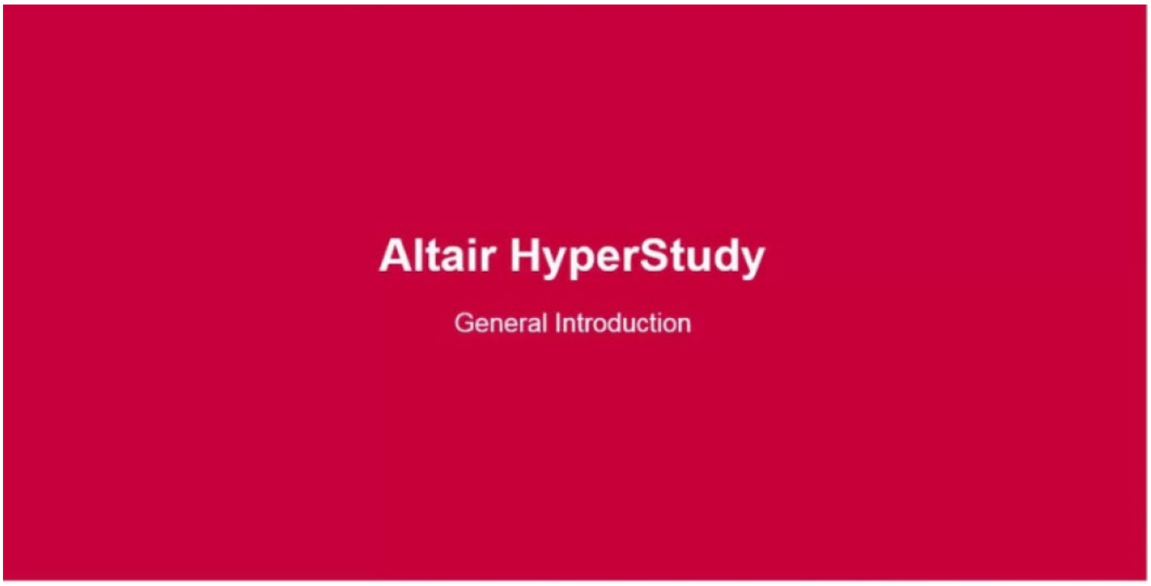
The general outline of HyperStudy for any study is as shown below:



The user sets up the study by identifying the variables, solvers for simulations and the responses to be monitored. Then the user can choose to perform either DOE / Optimization / Stochastic studies or perform any combination of them. The results can be processed using HyperStudy's variety of post processing utilities. The HyperStudy GUI equivalent of the flowchart is shown below.



Joe Pajot / Altair provides a very comprehensive overview won the workflow (process). In his video he even goes beyond the scope of this book: Fitting and Optimization.



https://altair-2.wistia.com/medias/hut0c4wz6b

In some details the working steps are:

6.6.1 Parametrization/Model Definition

Parametrization is a function of the model. Depending on the model you pick, whether it is HyperMesh, MotionView, Template, HyperStudy Internal Math, the parameters are imported through different mechanisms

The ease of connection depends on the model type:

- Workbench or Excel: Design Variables and Responses can be directly imported (see respective icon in the image below)
- HyperMesh/MotionView: provides a list of possible DVs, (HyperMesh model is the preferred model when there are shape variables).
- SimLab: SimLab has the advantage of CAD connectivity
- FEKO/Flux: Altair's electromagnetic solvers.
- Resource file: File with parameter definitions, e.g. Template file, HyperMesh file, Workbench project, …
- Solver input file: HyperStudy writes this file taking the information from the Resource file. Note: For some solvers such as RADIOSS the name specified as Solver input file must be the same as specified as Resource file. Moreover, for some solvers (like RADIOSS) two (or more files) need to be specified. These files need to be separated by ";". The file which is changing (includes model info) needs to be listed first.
- HyperStudy comes with the solver execution scripts for HyperWorks solvers, i.e. HyperStudy knows how to execute RADIOSS, OptiStruct, AcuSolve etc. For non-Altair solvers you may need to point to an existing script or you may need to create one first (see the Appendix "The Role of the Solver Script" for more details).

6.6.2 Define Design Variables

The information about the Design Variables is typically derived from the Resource file. Variable Data Type may be Real, Integer or even of type String and may be discrete / continuous.

Design Variables are either continuous or discrete.

Typical syntax of Design Variables

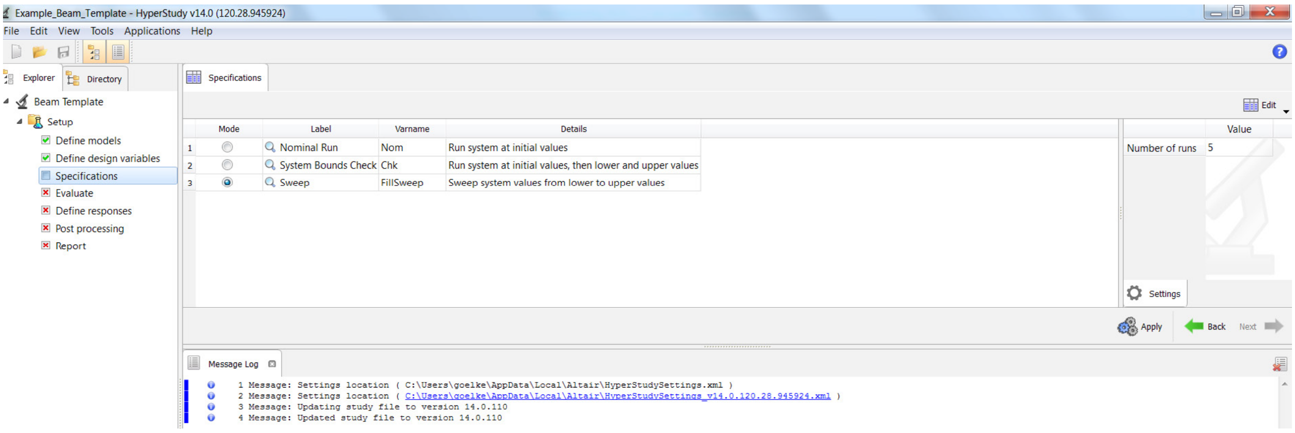
- Continuous: {parameter(varname_1, "Label 1", 1.0, 0.9, 1.1)}
- String:{parameter(varname_1, "comp 1", "0.0", "1.0","2.0", "3.0")}

A more advanced topic mentioned for completeness is Design Variable Linking (please check the Online Help for more information) as used for:

- Multi-disciplinary studies (you link the common variables so that they change consistently)
- Chained solvers → output of one solvers becomes input of next solver; → link the input of the second solver to the response of the first solver (tutorial 1060)

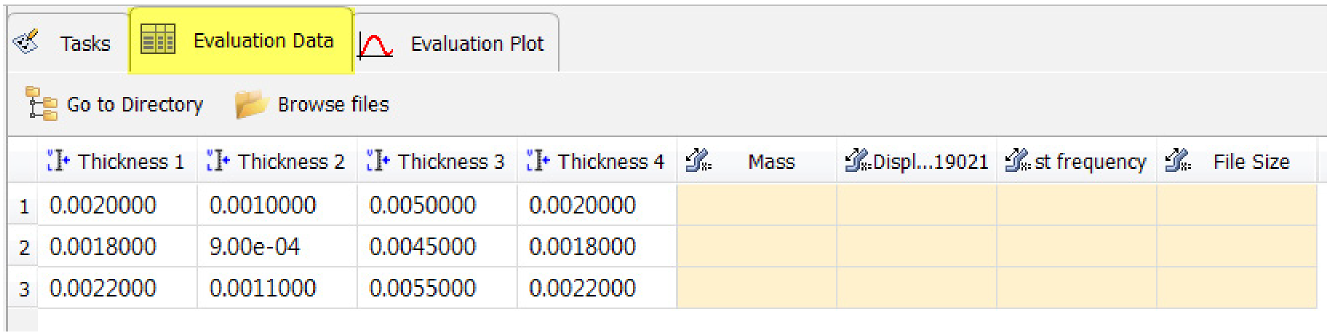
6.6.3 Specifications

After the design variables have been defined, we move forward to what is called “Specifications”



There are three options listed:

- Nominal Run – runs the initial design
- System Bounds Check- creates a table of 3 runs, one at the initial values, one at all lower bounds, one at all upper bounds

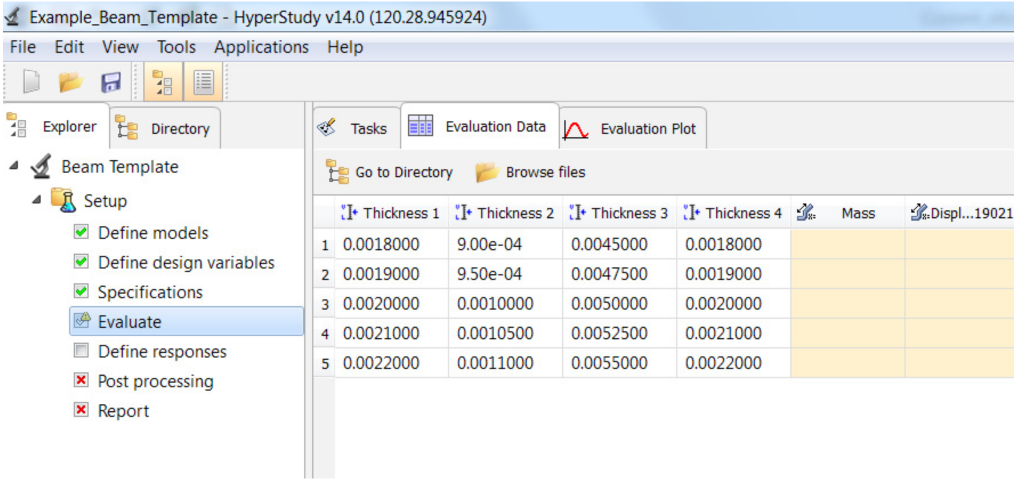


	Thickness 1	Thickness 2	Thickness 3	Thickness 4	Mass	Displ...19021	st frequency	File Size
1	0.0020000	0.0010000	0.0050000	0.0020000				
2	0.0018000	9.00e-04	0.0045000	0.0018000				
3	0.0022000	0.0011000	0.0055000	0.0022000				

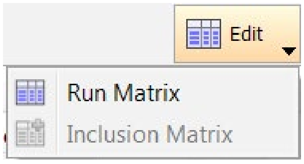
System Bounds Check. The combination of runs is automatically configured. To view this table, click on “Apply” and then on “Next”.

- Sweep

Creates a set of runs where the values of all the design variables vary from their lower to upper bound.



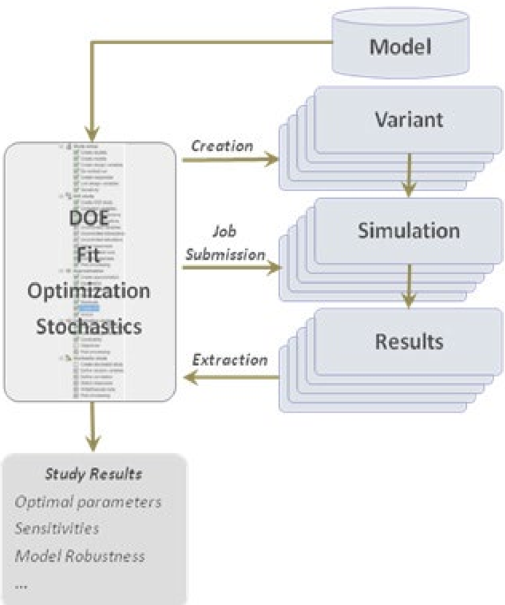
Of course, you can change the settings for lower and upper bounds according to your “needs” – just open the “Edit” panel and edit the Run Matrix.



Note: It is recommended to run, for instance the System Bound Checks to make sure that the model set-up is healthy/correct and behaves as expected. If three runs are too prohibitive, select two runs instead.

6.6.4 Evaluate / Solver Execution Script

The evaluation of a model consists of three phases: Writing, executing (Job Management), and Extracting.



In this section the focus is on Job Management (executing).

The role of the solver script maybe best described by “mapping mechanism”:

- A HyperStudy model maps independent variables to dependent variables

Simple question: Given an input file (e.g. HyperMesh, internal Math file) how do I generate output files?

This may seem simple, but answering this question is the blueprint. Imagine having an input file (like OptiStruct *.fem file):

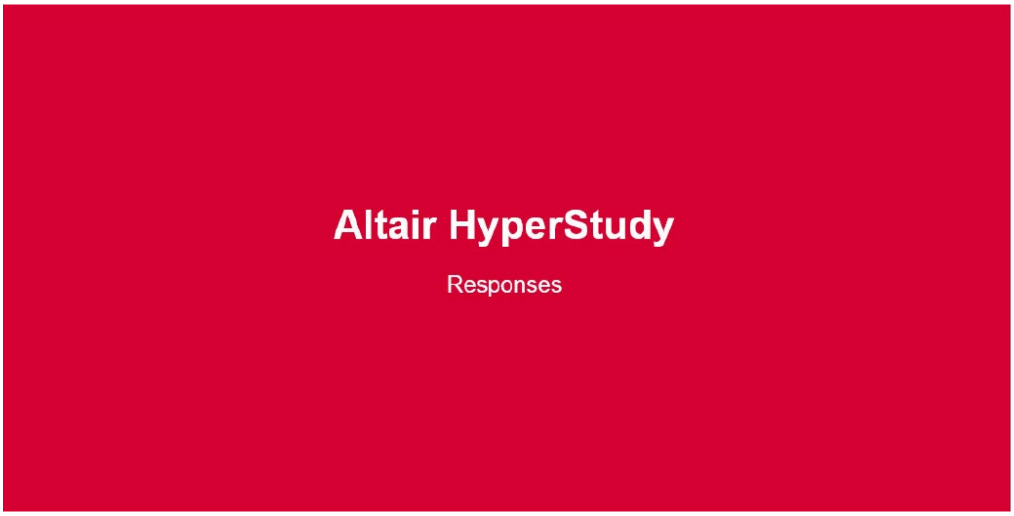
- How do you run it?

The solver script must do all which you would do manually to create the outputs given the input, but non-interactively, and should wait until completion before finishing.

Best Practice

- Understand the process to be executed
- Assuming you have input file, understand all the steps to get output files
- Make lists of steps
- Start writing a script outside of HyperStudy
- At this stage, stay simple and focus on basics
- Use hard coded paths
- Limit the arguments passed to the scrip
- Get the process running outside HyperStudy
- Once running, make limited adjustments to accept arguments (files, parameters)
- Test the script in HyperStudy’s nominal run
- If everything works, start adding environment variables
- Usually do this last, as it is the least crucial step but makes testing limited to HyperStudy only

6.6.5 Response Definition



https://altair-2.wistia.com/medias/djg879tj6q

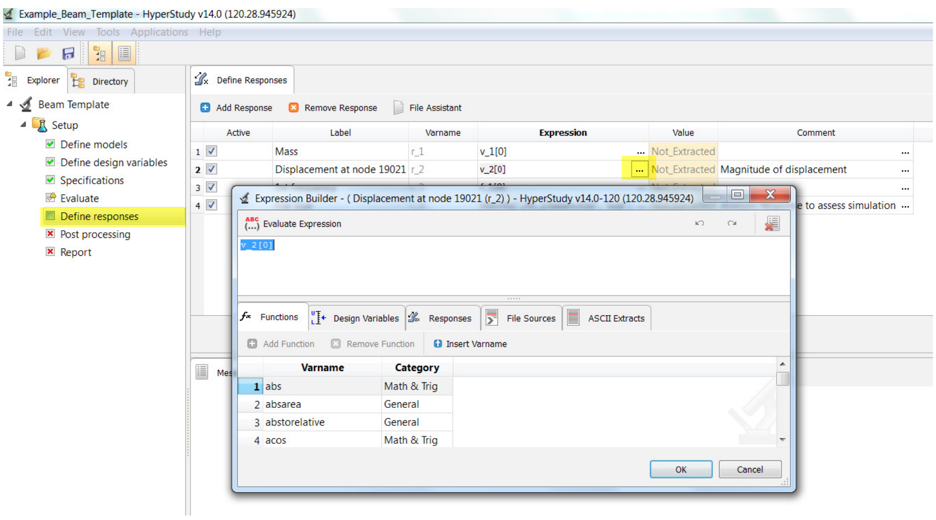
Question: What is a response?

Answer: Performance criteria that we are interested in

Something to note:

- Responses can be of varying complexity - sometimes it is just a number read from an output file, sometimes it is the outcome of a number of operations on the output of the solver
- We use the "Expression Builder" to construct these responses using simple pieces as building blocks.
- Expression builder has all the pieces to facilitate the construction of any expression, from simple to sophisticated.

The Expression Builder shown below can be accessed in the "Define response" step by clicking on the three dots … in the Expression column.



Expression builder

Moreover, a Response is usually a function of a solver output but sometimes it can be of only design variable values:

Response = volume = v_1[0]

Response = ratio of two dimensions = dv_1/dv_2

Next is a simple function of a solver output:

Response = abs (x-disp at node 1) = abs(v_1[0])

Next is a user-defined function of a solver output:

Response=area_2curves(v1x, v1x, v2x, v2y) where v1 and v2 are test and simulation data.

But it can even be just a number

Response = 10

(If you have an application where you are looking for a feasible design but don't have any responses to minimize or maximize, you can create such a response and make it the objective function)

Note: Data from a file (e.g. *.out file from OptiStruct) is read as a vector but Responses have to be scalars Hence, you have to perform an operation to create a scalar expression from a vector

Example:

Response = volume = v_1[0]

Why v_1[0]?

Volume is written to .out file

In this case, it is an OptiStruct *.out file and we have a reader for it.

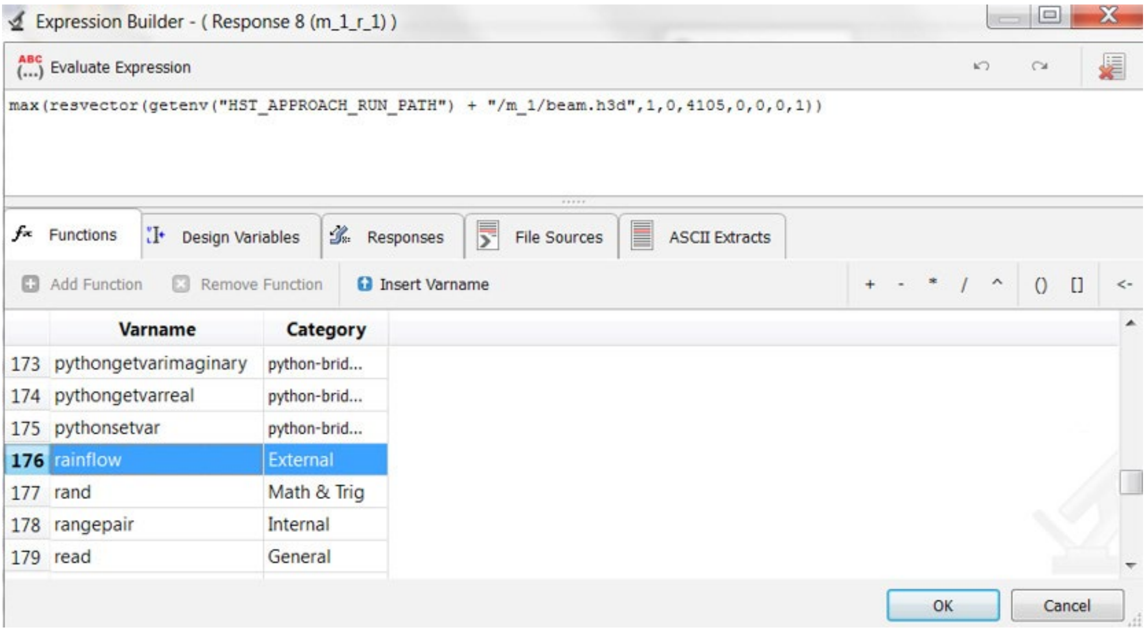
So, we use "File Source" to read it.

"File Source" reads it as a vector, v_1 of size 1.

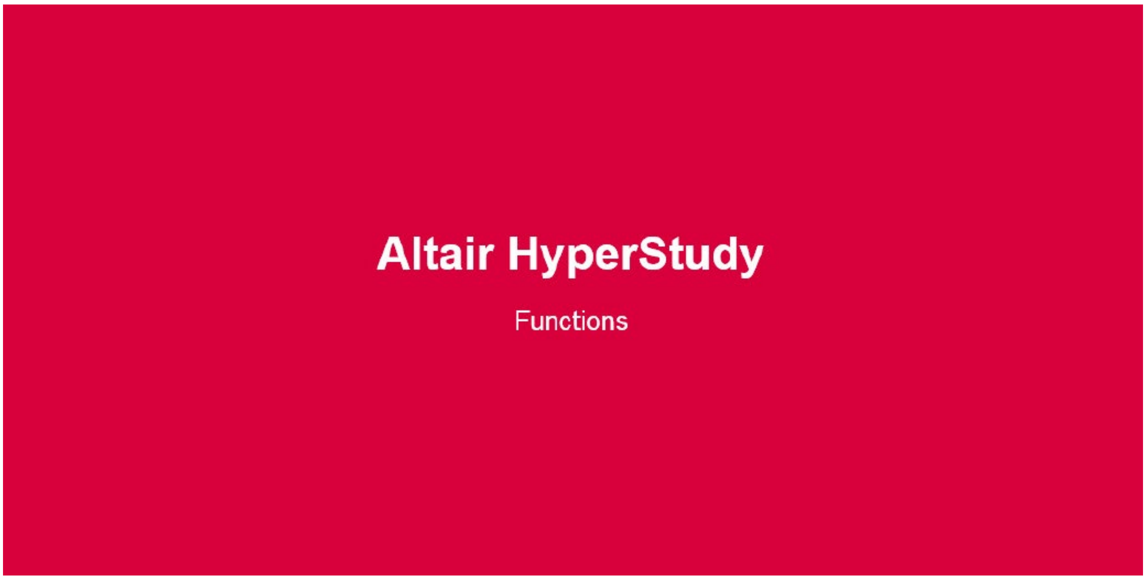
As we can only work with scalars in HyperStudy, and this particular response is a scalar, we put the index, v_1[0].

So, a response is an expression that is made of Functions, Design Variable, Responses, File Sources (for files that HyperWorks has readers for e.g. h3d, outfile), ASCII Extracts (for ASCII files HyperWorks has no reader for, e.g. an in-house-code).

Scalars, operators, design variables, responses, file outputs are the building blocks of a response (don't, miss to hit the INSERT Varname icon)



Functions

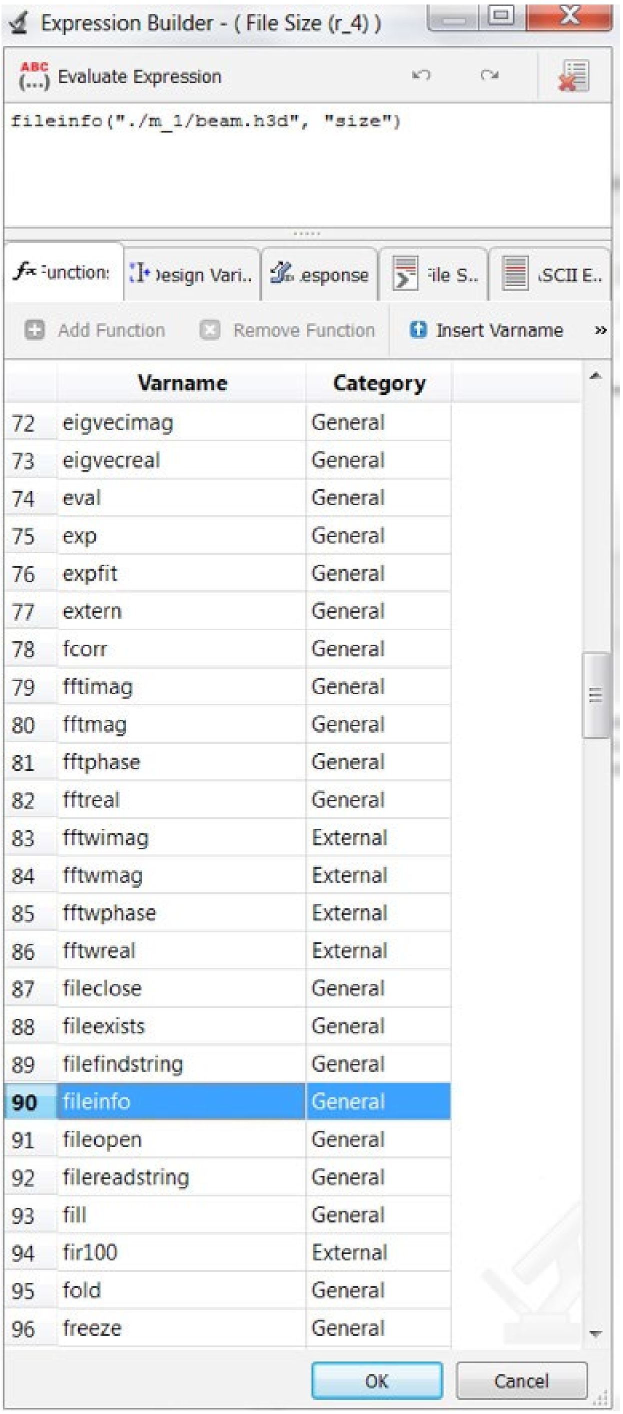


https://altair-2.wistia.com/medias/b2k1exi3n7

There are 250+ functions in the library, such as: Min, max, resvector, akima, butterworth, etc.

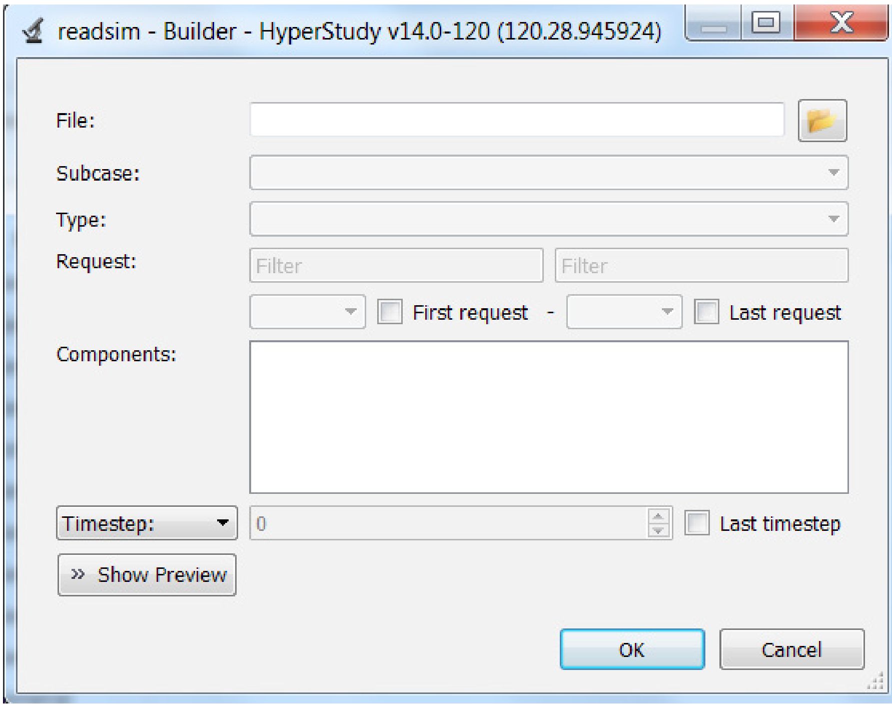
You can double click on the number next to the function varname to insert the function. This is particularly useful in resvector, readsim (details further below and Online Help).

In addition, you can add external functions which you may implement in Templex or HyperMath which can then be added as an external function. For more information about this option please see tutorials HS-1505: Register a Templex Function in HyperStudy and HS-1506: Register a HyperMath Function in HyperStudy.



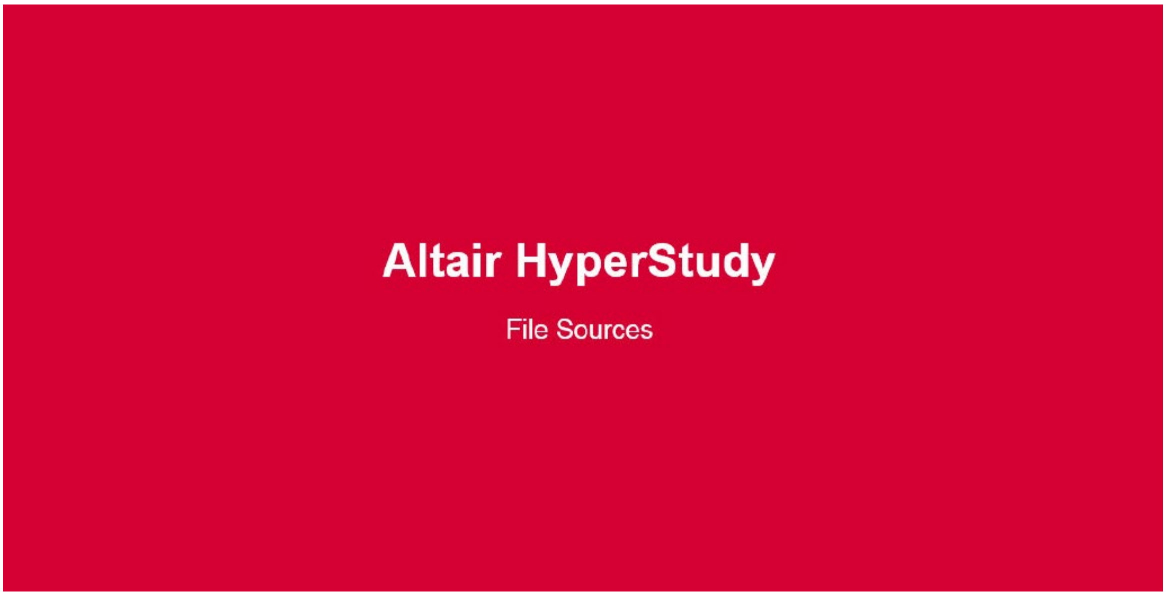
Some "Special" Functions

Readsim (available along with HyperStudy 14.0). Use case is to read the entire time history of the stress, which may then be used to identify its maximum value.



Double click on the number next to “Function→Readsim” opens up the above pop-up window.

File Sources (Vectors)

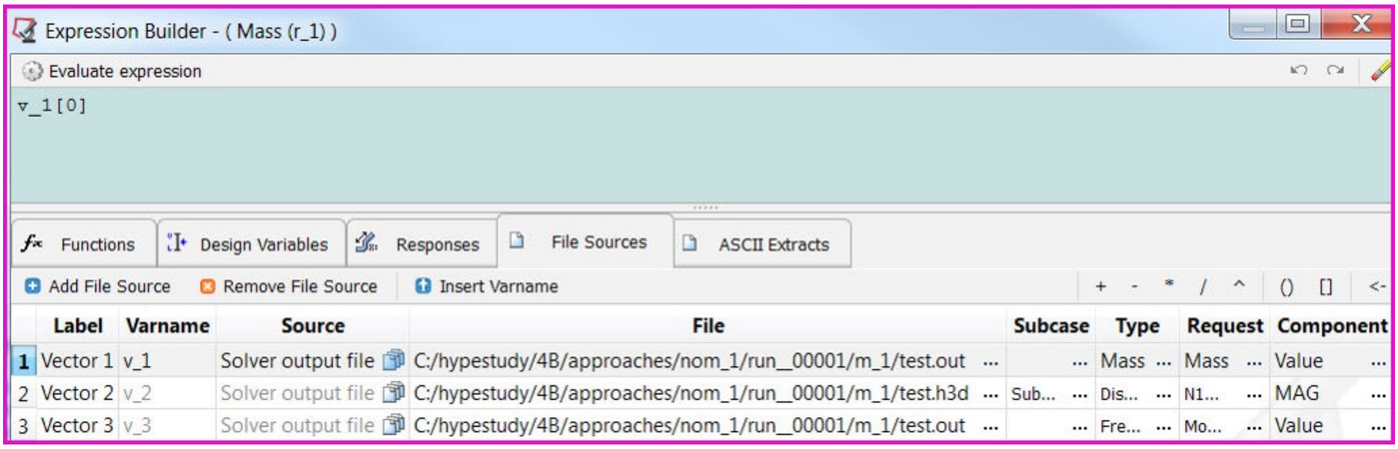


https://altair-2.wistia.com/medias/y66o28enan

File Sources allows for selection of result values from result files of solver runs for which HyperWorks has reader for.

In the pop-up window choose Type, Request and Component. Again note, that the results are stored as vectors. As the study requires a response to be a scalar value, use functions such as min(), max(), integral(),

define column with [..] e.g. v_1[0], v_1[numpts(v_1)-1] and recall that the index starts from 0.

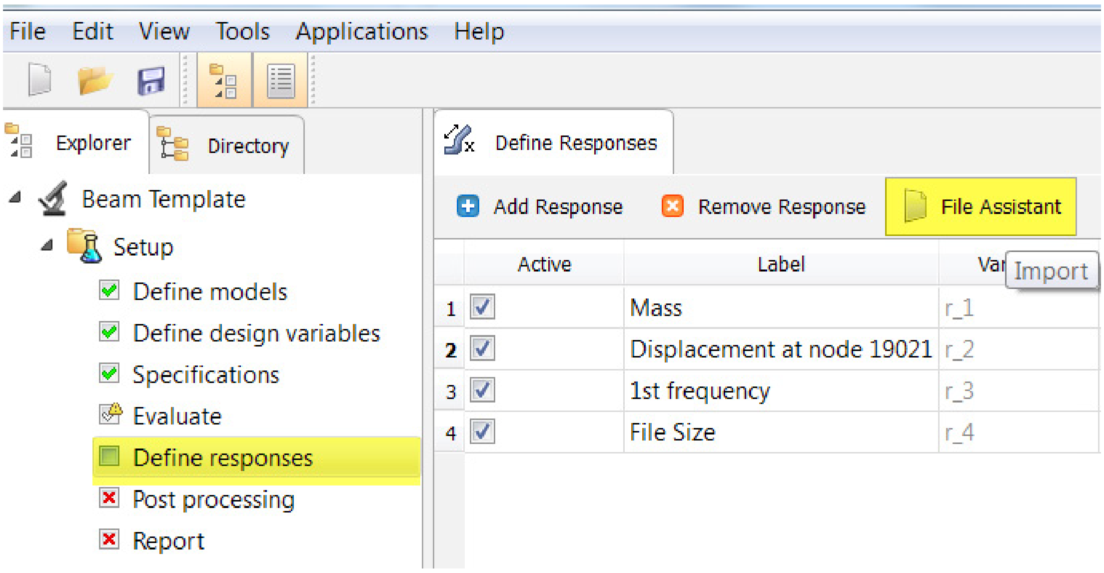


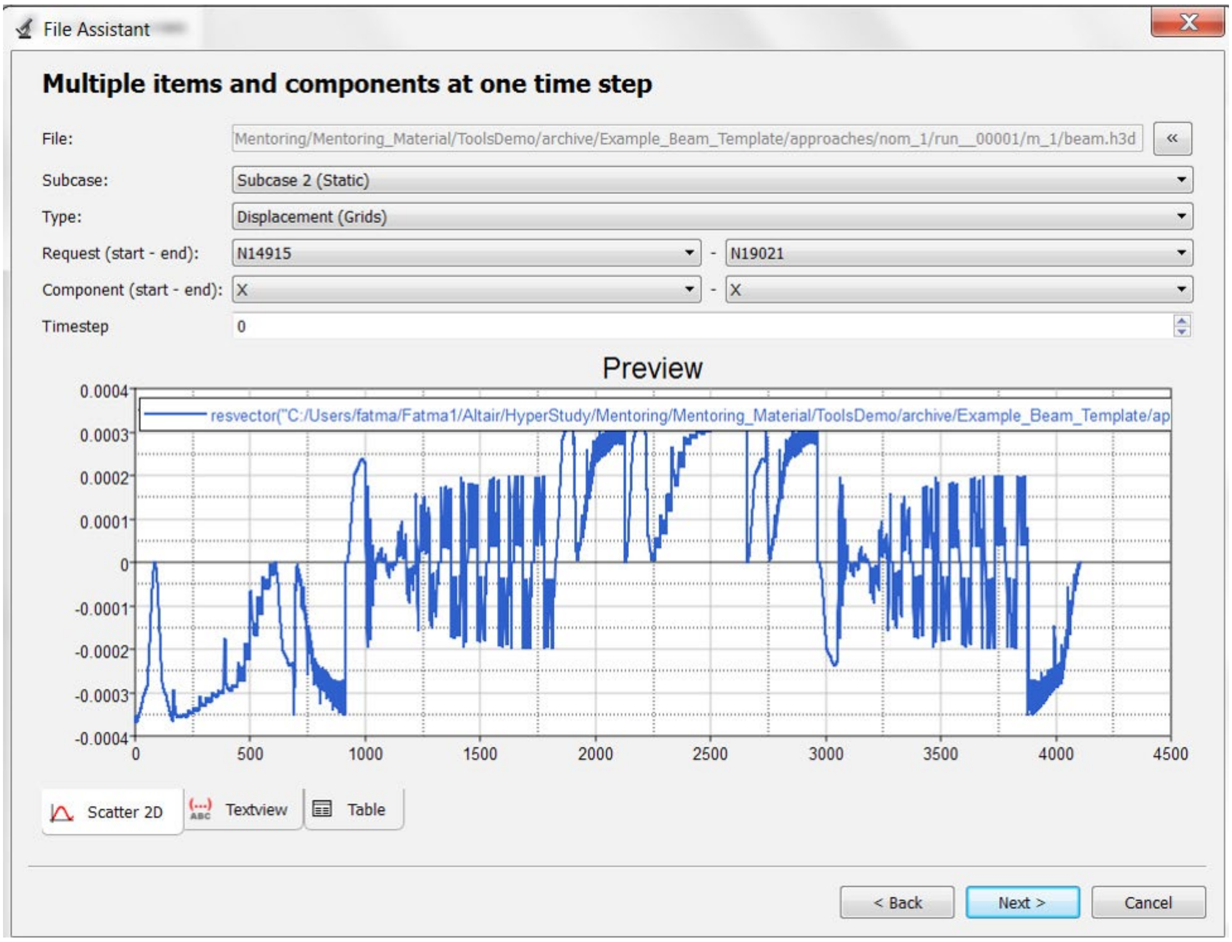
File Assistant

Along with HyperStudy 14.0 a new tool was introduced: File Assistant.

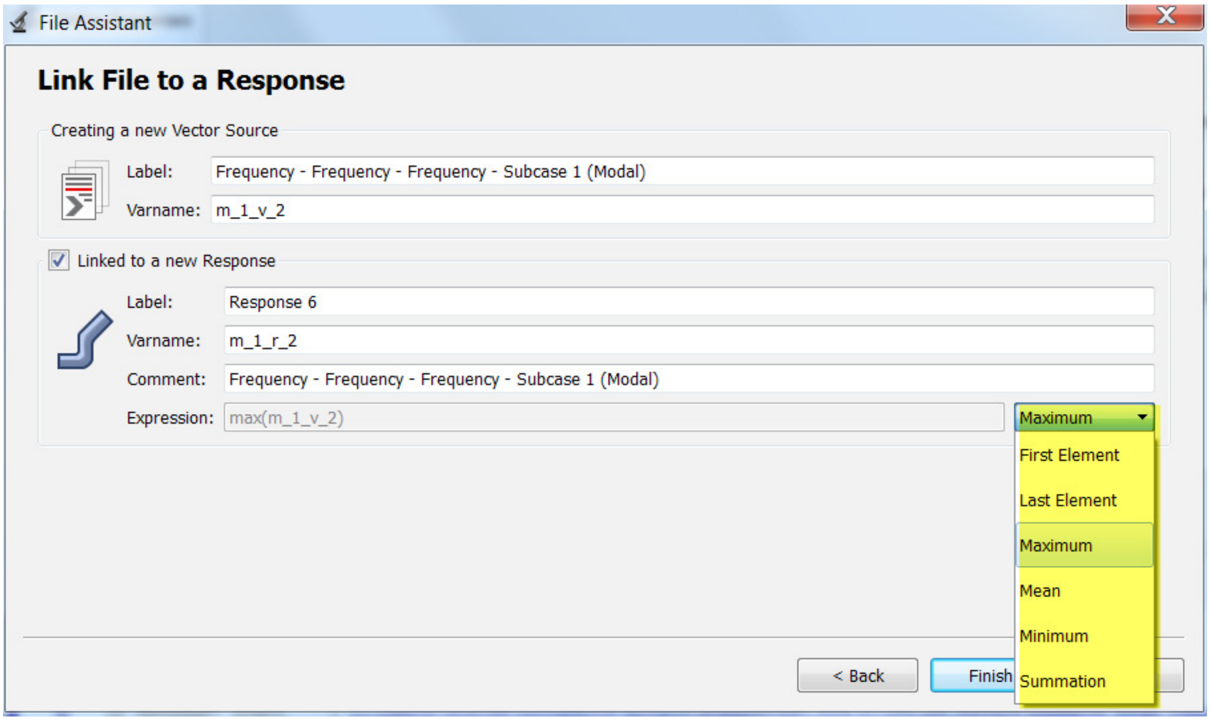
The Expression Builder is quite demanding. In many cases the Expression Builder can be replaced by the File Assistant. In the File Assistant UI, you select a single file and the type of data/result of interest (see image below).

The File Assistant can be accessed in the Define Response step →File Assistant.



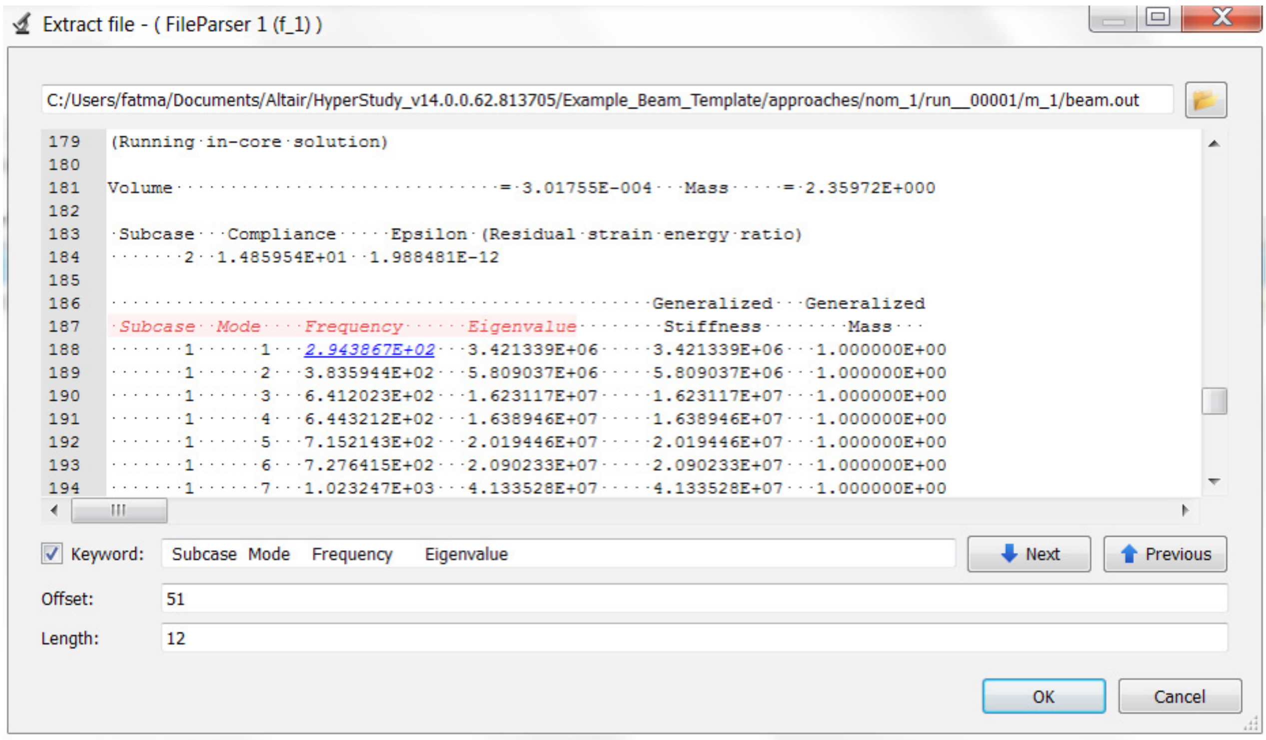


Within the File Assistant (which guides you through the various steps) you may also do some math on the results



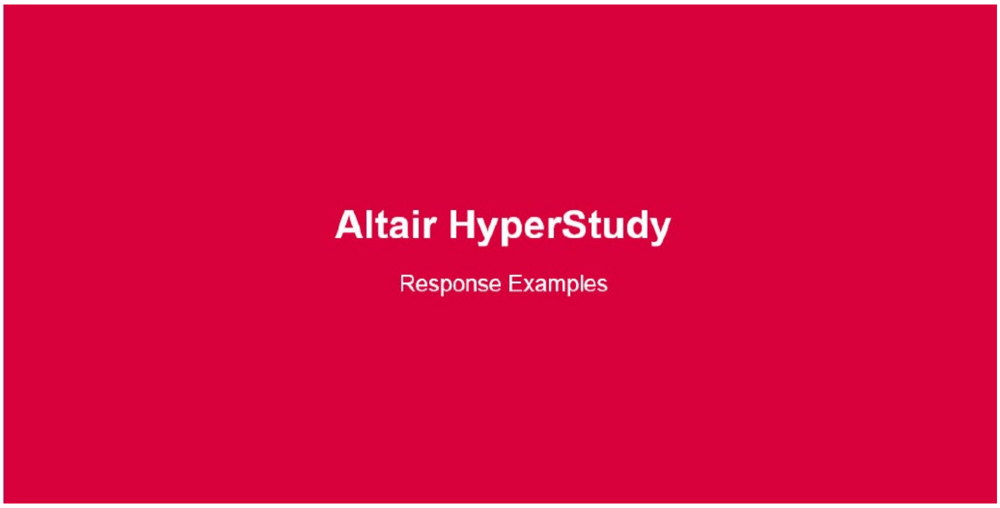
ASCII Extract

For ASCII files that we do not have readers for (ex: in-house code result files) make use of ASCII Extract



ASCII Extract provides the option to use a key word (here Subcase Mode Frequency Eigenvalue) to identify (or mark) the data of interest.

Response Examples

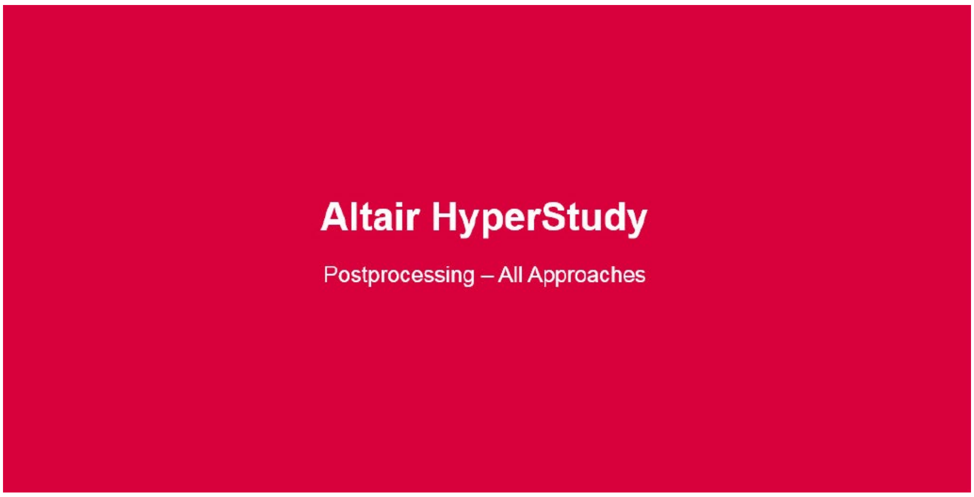


https://altair-2.wistia.com/medias/mpnjw4h5tf

We may face the situation that the output HyperStudy reads in is not in the same exact format we need. For instance, in crash analysis, you often want to look at the Force versus Displacement curve. However, most of the time you have Force vs. Time and Displacement versus Time.

If you run into a situation like this there are functions available that can help you, or contact the moderated Support Forum.

6.7 Postprocessing - All Approaches



https://altair-2.wistia.com/medias/v5k4js9k1j

Now after the DOE runs are completed probably the most demanding part of a DOE starts: Postprocessing.

There are several questions we need to answer:

What do I look at? How do I interpret the results? And what do I do with that newly gained piece of information?

The following table (or overview) may be of help:

	Doe	Fit	Optimization	Stochastic	Comments
Anova		X			Identify the importance of LSR factors.
Diagnostics		X			Assess response surface quality in summary.
Distribution	X	X	X	X	Visualize data trends and identify outliers.
Linear Effects	X				Identify important variables.
Integrity	X	X	X	X	Review statistics of data sets.
Interactions	X				Identify interconnection between variables.
Iterations			X		Visualize scatter history of optimizations.
Optima			X		Visualize Pareto Frontiers.
Parallel Coordinate	X	X	X	X	Identify trends in large data sets.
Pareto Plot	X				Plot the effects of variables on responses in hierarchical order (highest to lowest).
Reliability				X	Lookup specific reliability values.
Reliability Plot				X	Visualize the representation of system reliability.
Residuals		X			Asses response surface quality in detail.
Scatter 2D	X	X	X	X	Visualize data sets in 2 dimensions.
Scatter 3D	X	X	X	X	Visualize data sets in 3 dimensions.
Summary	X	X	X	X	Present raw data in tabular form.
Trade-Off		X			Interactive response suface tool to perform "what if" scenarios.

In HyperStudy there are 6 fundamental ways of looking/postprocessing data – regardless whether it is from DOE or Optimization.

These options are depicted on the left side of the image above:

Integrity, Summary, Parallel Coordinate, Distribution, Scatter 2D, Scatter3D

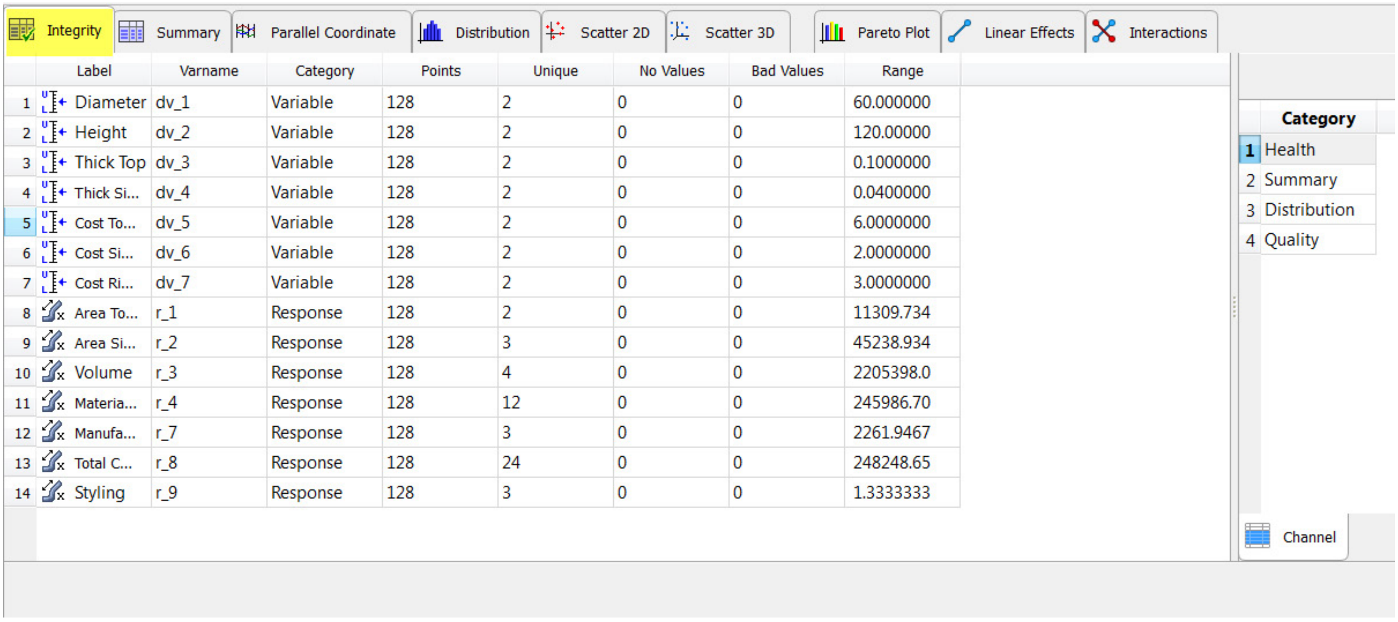
The DOE specific postprocessing options (highlighted by the green box) Pareto Plot, Linear Effects, and Interactions are discussed further below.

The Integrity Tab contains a series of statistical measures on variables and responses:

- Health High level summary of statistics used to easily spot inconsistent, non-changing, or missing data
- Summary Basic descriptive statistics that presents information on the data in groups such as quartiles or ranges
- Distribution Detailed descriptive statistics used to quantitatively describe the distribution of data points
- Quality Values typically used in Quality Engineering

6.7.1 Integrity – Category Health Check

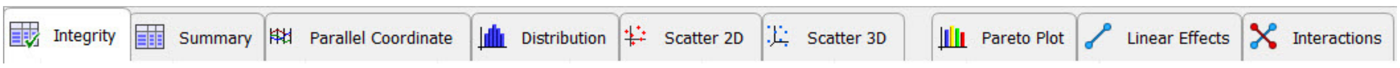
The Health Check provides a high-level summary of statistics used to easily spot inconsistent, non-changing, or missing data.

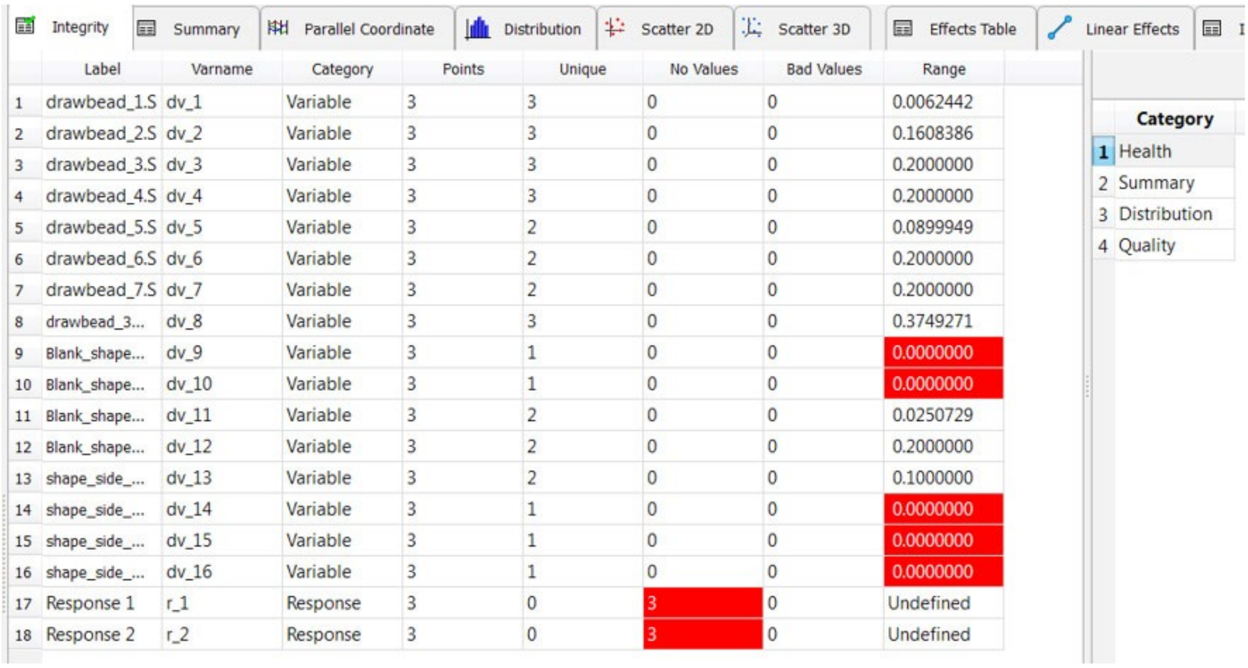


	Label	Varname	Category	Points	Unique	No Values	Bad Values	Range
1	Diameter	dv_1	Variable	128	2	0	0	60.000000
2	Height	dv_2	Variable	128	2	0	0	120.00000
3	Thick Top	dv_3	Variable	128	2	0	0	0.1000000
4	Thick Si...	dv_4	Variable	128	2	0	0	0.0400000
5	Cost To...	dv_5	Variable	128	2	0	0	6.0000000
6	Cost Si...	dv_6	Variable	128	2	0	0	2.0000000
7	Cost Ri...	dv_7	Variable	128	2	0	0	3.0000000
8	Area To...	r_1	Response	128	2	0	0	11309.734
9	Area Si...	r_2	Response	128	3	0	0	45238.934
10	Volume	r_3	Response	128	4	0	0	2205398.0
11	Materia...	r_4	Response	128	12	0	0	245986.70
12	Manufa...	r_7	Response	128	3	0	0	2261.9467
13	Total C...	r_8	Response	128	24	0	0	248248.65
14	Styling	r_9	Response	128	3	0	0	1.3333333

Integrity Health Check

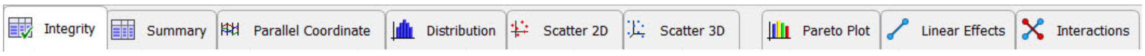
In the image above we see a 100 run DOE which delivered in general 100 unique (result/response; in here some results must have been the same, i.e. indicated by 99/98) points. There are no entries without any values (No Values), and there are also no "Bad Values" (e.g. divided by zero)





	Label	Varname	Category	Points	Unique	No Values	Bad Values	Range
1	drawbead_1.S	dv_1	Variable	3	3	0	0	0.0062442
2	drawbead_2.S	dv_2	Variable	3	3	0	0	0.1608386
3	drawbead_3.S	dv_3	Variable	3	3	0	0	0.2000000
4	drawbead_4.S	dv_4	Variable	3	3	0	0	0.2000000
5	drawbead_5.S	dv_5	Variable	3	2	0	0	0.0899949
6	drawbead_6.S	dv_6	Variable	3	2	0	0	0.2000000
7	drawbead_7.S	dv_7	Variable	3	2	0	0	0.2000000
8	drawbead_3...	dv_8	Variable	3	3	0	0	0.3749271
9	Blank_shape...	dv_9	Variable	3	1	0	0	0.0000000
10	Blank_shape...	dv_10	Variable	3	1	0	0	0.0000000
11	Blank_shape...	dv_11	Variable	3	2	0	0	0.0250729
12	Blank_shape...	dv_12	Variable	3	2	0	0	0.2000000
13	shape_side_...	dv_13	Variable	3	2	0	0	0.1000000
14	shape_side_...	dv_14	Variable	3	1	0	0	0.0000000
15	shape_side_...	dv_15	Variable	3	1	0	0	0.0000000
16	shape_side_...	dv_16	Variable	3	1	0	0	0.0000000
17	Response 1	r_1	Response	3	0	3	0	Undefined
18	Response 2	r_2	Response	3	0	3	0	Undefined

In this example there are 3 responses which hadn't any values, and there is a number of ranges that are "zero" indicating that nothing has been changed. Hence, a warning (colored in red) is prompted requesting us to go back to the date to better understand what happened. For instance, the data shown in the table may come from a test and there haven't been 3 test data (No Values) available. As a consequence, the input data must be cleaned up first; or it may have been simulation data and the simulation run didn't converge.



Integrity | Summary | Parallel Coordinate | Distribution | Scatter 2D | Scatter 3D | Optima | Iterations

	Label	Varname	Category	Points	Unique	No Values	Bad Values	Range
1	drawbead_1.S	dv_1	Variable	50	16	0	0	0.1867184
2	drawbead_2.S	dv_2	Variable	50	10	0	0	0.4120161
3	drawbead_3.S	dv_3	Variable	50	11	0	0	0.5000000
4	drawbead_4.S	dv_4	Variable	50	13	0	0	0.5024482
5	drawbead_5.S	dv_5	Variable	50	14	0	0	0.2000768
6	drawbead_6.S	dv_6	Variable	50	6	0	0	0.3000000
7	drawbead_7.S	dv_7	Variable	50	7	0	0	0.3000000
8	drawbead_3...	dv_8	Variable	50	17	0	0	0.6000000
9	Blank_shape...	dv_9	Variable	50	18	0	0	0.2000000
10	Blank_shape...	dv_10	Variable	50	6	0	0	0.3000000
11	Blank_shape...	dv_11	Variable	50	18	0	0	0.6000000
12	Blank_shape...	dv_12	Variable	50	6	0	0	0.3000000
13	shape_side_...	dv_13	Variable	50	12	0	0	0.1650000
14	shape_side_...	dv_14	Variable	50	13	0	0	0.2000000
15	shape_side_...	dv_15	Variable	50	5	0	0	0.3000000
16	shape_side_...	dv_16	Variable	50	7	0	0	0.3000000
17	Response 1	r_1	Response	50	26	0	0	30197.126
18	Response 2	r_2	Response	50	25	0	0	71.191765
19	Objective 1	obj_1	Objective	50	26	0	0	30197.126
20	Constraint 1	c_1	Constraint	50	25	0	0	71.191765

	Category
1	Health
2	Summary
3	Distribution
4	Quality

Channel

Integrity Health Check

In this table we see that 50 runs have been conducted. However, we only see 25 (26) unique results. So why do we run that many simulations if we get only 25 results at the end. Again, this indicates that something may be wrong.

6.7.2 Integrity – Category Summary Check

This check provides basic descriptive statistics that presents information on the data in groups such as quartiles or ranges

Mean = average, first statistical moment

Median = " middle" of a sorted list of numbers)

1st quartile = 25th percentile (splits off the lowest 25% of data from the highest 75%)

3rd quartile = 75th percentile (splits off the highest 25% of data from the lowest 75%)

sort	1	2	3	4	5	6	7	8	9	10	11	12	13
array	3	4	5	7	8	9	15	17	18	19	20	21	25
mean	13.2												
media	15												
1st Q	7												
2nd Q	15												
3rd Q	19												
4th Q	25												

The simple Excel sheet gives an overview on relevant / important definitions.

Integrity | Summary | Parallel Coordinate | Distribution | Scatter 2D | Scatter 3D | Pareto Plot | Linear Effects | Interactions

	Label	Varname	Category	Minimum	1st Qu	Median	Mean	3rd Qu	Maximum	Range
1	Diameter	dv_1	Variable	30.000000	30.000000	60.000000	60.000000	90.000000	90.000000	60.000000
2	Height	dv_2	Variable	60.000000	60.000000	120.00000	120.00000	180.00000	180.00000	120.00000
3	Thick Top	dv_3	Variable	0.2000000	0.2000000	0.2500000	0.2500000	0.3000000	0.3000000	0.1000000
4	Thick Si...	dv_4	Variable	0.1000000	0.1000000	0.1200000	0.1200000	0.1400000	0.1400000	0.0400000
5	Cost To...	dv_5	Variable	2.0000000	2.0000000	5.0000000	5.0000000	8.0000000	8.0000000	6.0000000
6	Cost Si...	dv_6	Variable	1.0000000	1.0000000	2.0000000	2.0000000	3.0000000	3.0000000	2.0000000
7	Cost Ri...	dv_7	Variable	1.5000000	1.5000000	3.0000000	3.0000000	4.5000000	4.5000000	3.0000000
8	Area To...	r_1	Response	1413.7167	1413.7167	7068.5835	7068.5835	12723.450	12723.450	11309.734
9	Area Si...	r_2	Response	5654.8668	11309.734	16964.600	22619.467	33929.201	50893.801	45238.934
10	Volume	r_3	Response	84823.002	169646.00	508938.01	848230.02	1526814.0	2290221.0	2205398.0
11	Materia...	r_4	Response	8482.3002	24033.184	57962.384	80581.852	135716.80	254469.00	245986.70
12	Manufa...	r_7	Response	282.74334	565.48668	848.23002	1130.9734	1696.4600	2544.6900	2261.9467
13	Total C...	r_8	Response	8765.0435	24598.670	58527.871	81712.825	137413.26	257013.69	248248.65
14	Styling	r_9	Response	0.1666667	0.3333333	0.5000000	0.6666667	1.0000000	1.5000000	1.3333333

Category
1 Health
2 Summary
3 Distribution
4 Quality

Channel

All the information from above such as mean, median etc. goes into BoxPlots (under Distribution) as shown further below.

6.7.3 Integrity – Category Distribution Check

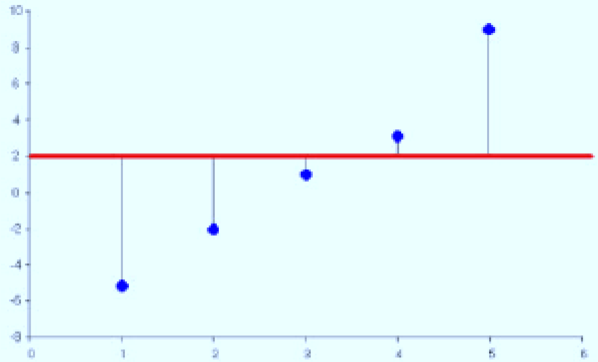
Integrity | Summary | Parallel Coordinate | Distribution | Scatter 2D | Scatter 3D | Pareto Plot | Linear Effects | Interactions

	Label	Varname	Category	Variance	Std. Dev.	Avg. Dev.	CoV.	Skewness	Kurtosis	RMS
1	Diameter	dv_1	Variable	907.08661	30.117879	30.000000	0.5019646	0.0000000	-2.0320000	67.082039
2	Height	dv_2	Variable	3628.3465	60.235757	60.000000	0.5019646	0.0000000	-2.0320000	134.16408
3	Thick Top	dv_3	Variable	0.0025197	0.0501965	0.0500000	0.2007859	5.62e-13	-2.0320000	0.2549510
4	Thick Si...	dv_4	Variable	4.03e-04	0.0200786	0.0200000	0.1673215	-1.10e-13	-2.0320000	0.1216553
5	Cost To...	dv_5	Variable	9.0708661	3.0117879	3.0000000	0.6023576	0.0000000	-2.0320000	5.8309519
6	Cost Si...	dv_6	Variable	1.0078740	1.0039293	1.0000000	0.5019646	0.0000000	-2.0320000	2.2360680
7	Cost Ri...	dv_7	Variable	2.2677165	1.5058939	1.5000000	0.5019646	0.0000000	-2.0320000	3.3541020
8	Area To...	r_1	Response	3.22e+07	5677.0864	5654.8668	0.8031434	2.05e-15	-2.0320000	9052.2036
9	Area Si...	r_2	Response	2.90e+08	17031.259	14137.167	0.7529470	0.8994640	-0.7991817	28274.334
10	Volume	r_3	Response	7.61e+11	872592.37	720995.51	1.0287214	0.9028666	-0.8907624	1214480.6
11	Materia...	r_4	Response	4.80e+09	69312.203	56725.382	0.8601466	1.0650398	0.2166536	106113.54
12	Manufa...	r_7	Response	725159.47	851.56296	706.85835	0.7529470	0.8994640	-0.7991817	1413.7167
13	Total C...	r_8	Response	4.86e+09	69734.636	57078.812	0.8534111	1.0555325	0.1913818	107246.97
14	Styling	r_9	Response	0.2519685	0.5019646	0.4166667	0.7529470	0.8994640	-0.7991817	0.8333333

Category
1 Health
2 Summary
3 Distribution
4 Quality

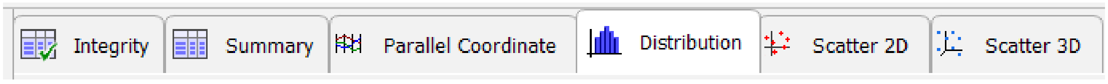
Channel

Some general explanations regarding the different columns is shown under “Distributions”. More information about the statistical indicators for responses are listed in the Integrity table columns.

Column	**Description**
Average deviation	Evaluated using the following expression: $$\frac{\sum_{i=1}^{N}\lvert x_i - \bar{x}\rvert}{N}$$  The horizontal line in the above plot represents the average of the values in the vector. The vertical lines represent the differences between the values of the vector and the average of the values. The average deviation is the average difference between the vector elements and the average of the vector elements. The sign of each element is not taken into consideration when calculating the deviation. The sign of each element is taken into consideration when calculating the average of the elements.
Coefficient of Variation (COV)	A measure of relative dispersion given by: $$CoV = \frac{Standard\ deviaiton}{Mean}$$ The use of variation lies partly in the fact that the mean and standard deviation tend to change together in many experiments. The higher the CoV, the higher the variability. The lower the CoV, the less the variability of the data. CoV is seldom of interest where the mean is likely to be near zero.
Kurtosis	Measure of flatness of a distribution.

Standard Deviation	This is the square root of the variance. It is a commonly used measure of dispersion, i.e. how spread the data is. For instance, if it is a response and it is spread too much it indicates that it is not a robust design
Skewness	Indicates whether the probability distribution is skewed to the right or to the left. If the skewness is zero, the probability distribution is symmetric about the mean of the distribution. If the skewness is less than zero, the probability distribution is skewed to the left of the mean of the distribution. If the skewness is greater than zero, the probability distribution is skewed to the right of the mean of the distribution. For instance, if we look at stress and if it is left inclined it is quite admirable as it means there a more points on the lower stress values
RMS	Calculates the square root of the mean of the sum of the squares of all response values using the following equation: $\sqrt{\frac{\sum x_i^2}{N}}$
Variance	Evaluated using the following expression: $\frac{\sum_{i=1}^{N}(x_i - \bar{x})^2}{N-1}$

6.7.4 Distribution (Main Tab)



In the Distribution tab, view all variable and response data in a histogram or box plot. Switch between histogram and box plot view by clicking the (histogram) or (box plot) buttons located above the Channel selector.

Histograms

In Histogram view, the following data is displayed for all variables and responses

Histograms display the frequency of runs in a sub-range of response values. The size of the sub-range is defined as the total range of the response value, divided by the number of bins. Histograms are displayed by red bins

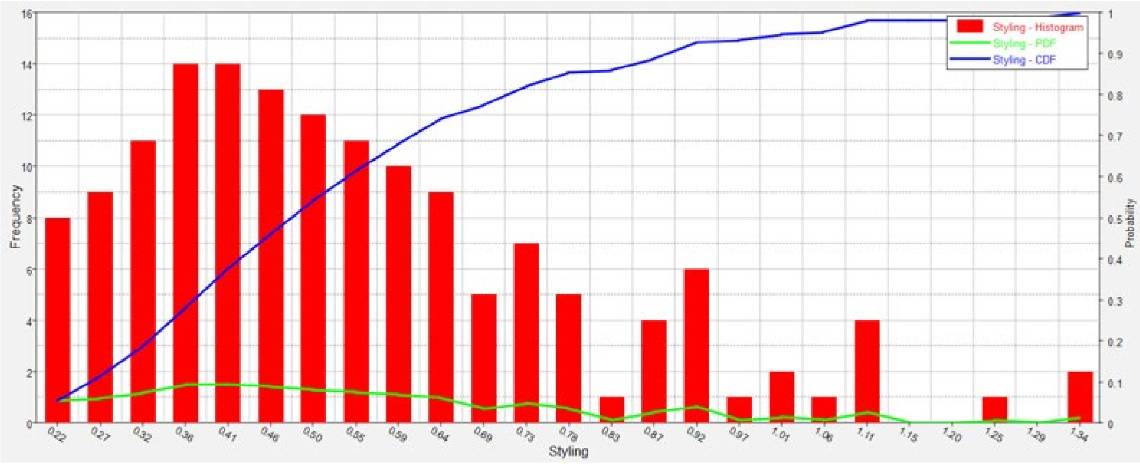
PDF (Probability Density Function) curves illustrate the probability of the response being equal to a particular value. PDF is displayed as a green curve.

- CDF (Cumulative Density Function) curves illustrate the probability of the response being less than or equal to a particular value.

CDF is displayed as a blue curve.

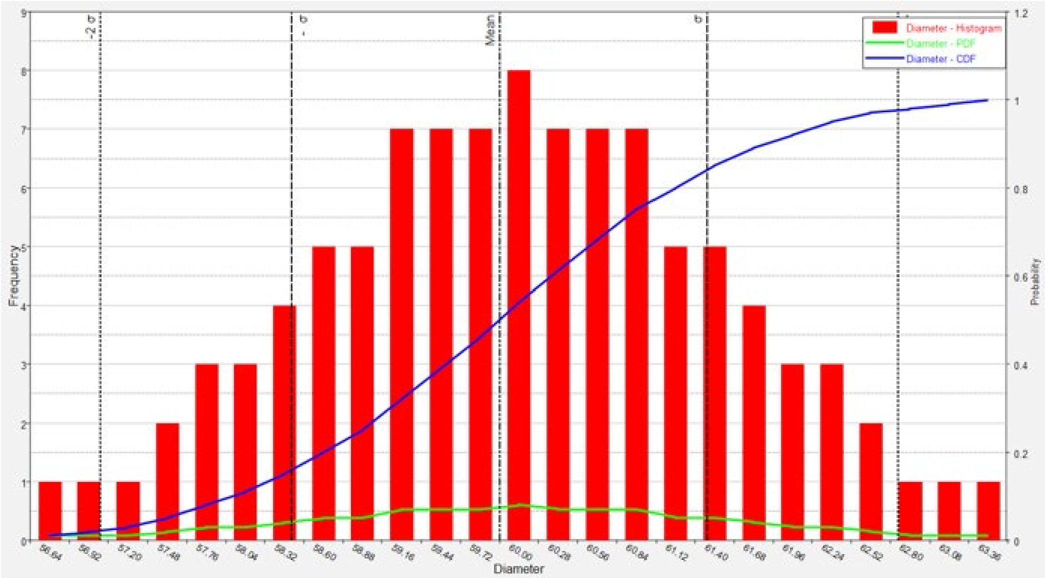
Or in other words - It tells you how many peaks there are, what kind of distribution it follows etc.:

- displays the frequency of an entity value within a bin
- out of 150 runs, 8 had a styling value in the 0.22 bin, 14 had a styling value in the 0.36 bin, etc. (see image below)
- gives information about the shape/properties (peaks, skewness, modality (bimodal, etc.), kurtosis) of the distribution.
- gives the range of values.



PDF refers to “Probability of Distribution Function” which runs – more or less - parallel to the Histogram

CDF is the sum of the PDF’s – it goes from zero to one, because you sum these values up. Its slope tells you where changes occur faster or slower.



Box Plots

In Box Plot view, data is sorted, and a box is drawn from the lower quartile (1st quartile, Q1, 25%) to the upper quartile (3rd quartile, Q3, 75%). Quartiles of a sorted data set consist of the three points (Q1, Q2 which is also the median, and Q3) that divide the data set into four groups, each group comprising a quarter of the data. The median and mean of the data are also marked in the box. In HyperStudy, this box is painted dark green (see images below).

Box plots may also have lines extending vertically from the box to indicate the data outside the lower and upper quartiles.

Furthermore, to identify outliers, these lines may extend only to the “whiskers” as opposed to the minimum and maximum of the data.

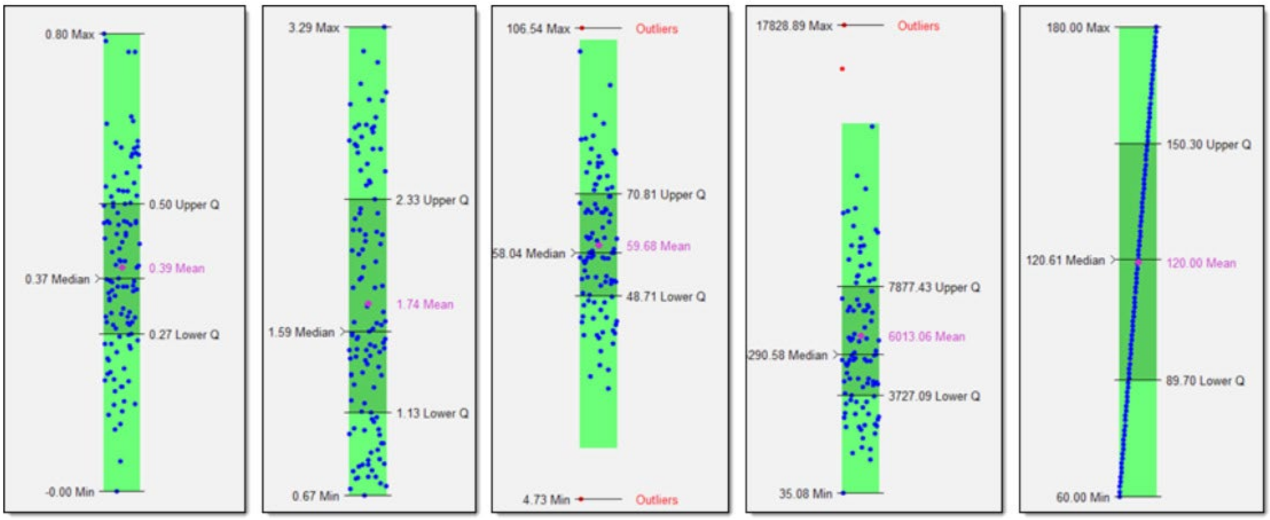
(Outliers refer to designs which behave much differently than other designs. Outliers may behave much better or worse than other designs. Hence, there is a need to investigate the causes of the outliers so that you either better understand which combination of design variables caused the much better performance, and vice versa, what should be avoided to end up with a poorly performing design).

Whisker location is calculated as a function of lower and upper quartile and the difference between them (this difference is known as interquartile range, IQR) as:

Lower whisker = Q1 – 1.5*IQR

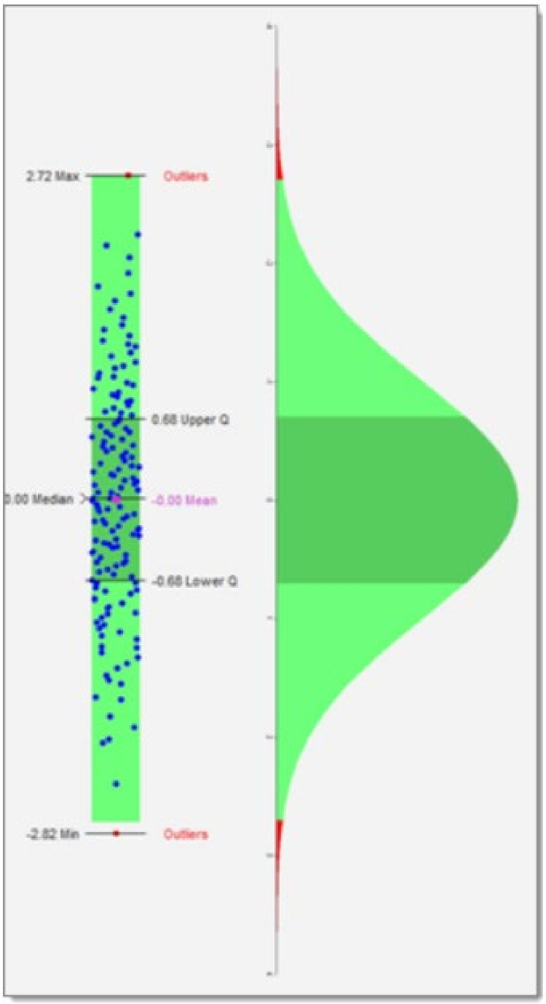
Upper whisker = Q3 + 1.5*IQR

Any data that is not within the whiskers are identified as “outliers”. In HyperStudy, whiskers are displayed as a light green box instead of as a vertical line, and data points are indicated by blue dots. Horizontal scale is their run number and vertical scale is their value.

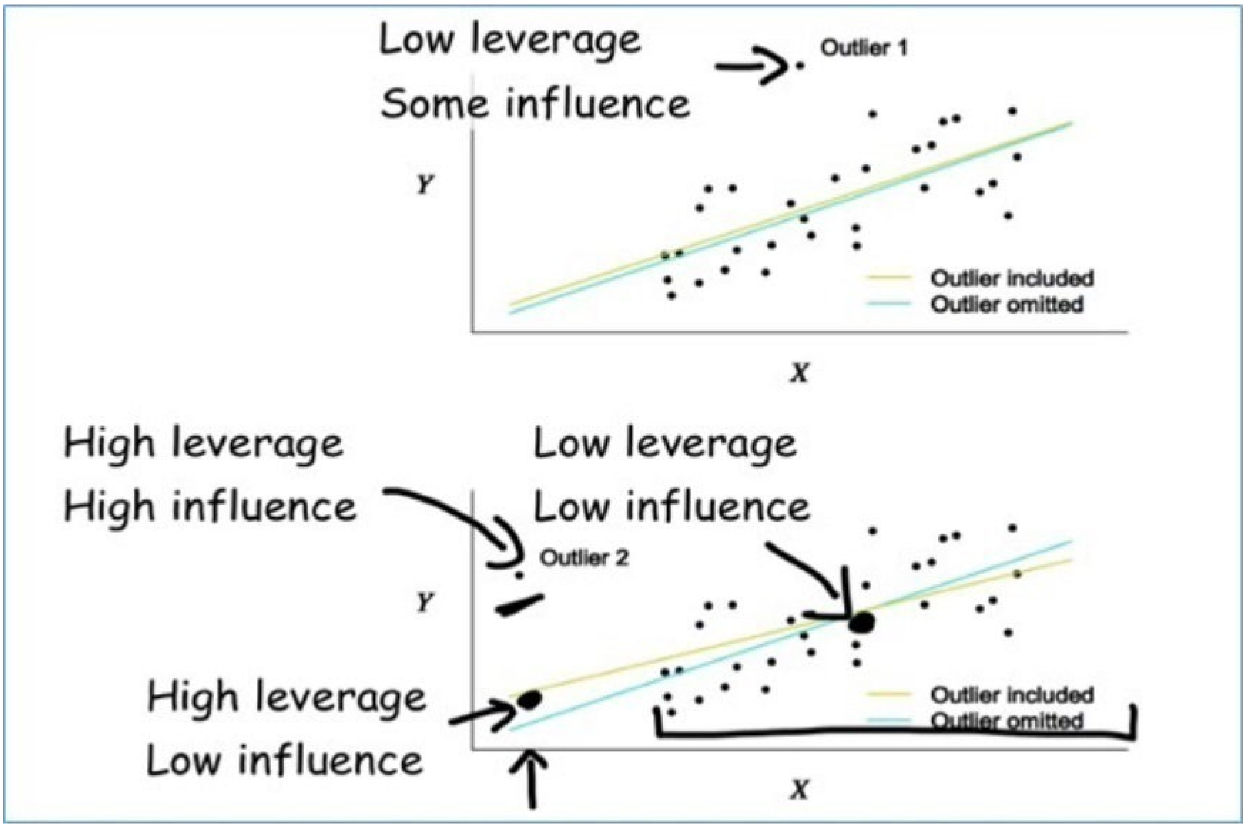


Box plots display the distribution of data. Use box plots to find the range, mean, median, quartiles, whiskers and outliers. This information tells you the spread and skewness of the data and helps you identify outliers. It is important that you understand the spread and skewness in order to understand and improve the variations in the data. Identifying the outliers gives you an opportunity to investigate these data points and resolve possible issues that you may have missed.

The image below illustrates a comparison of a box plot of data sampled from a normal distribution to the theoretical probability distribution function of the normal distribution. The dark green color indicates the interquartile range, the Light green color indicates the range of the whiskers, and the red color indicates outliers.



So, why should we care about outliers? Take a look at image below. The top plot shows a data value that looks like an Outlier (Outlier 1). Using a linear least square estimate one can see effect of including and omitting Outlier 1. Observe that the two lines are quite close to each other, indicating that the outlier has very less influence on the predictive nature of fit. In the lower plot however, this is not the case. Outlier 2 when included exerts more influence and pulls the best fitted line towards it. The difference between two cases is that outlier 2 has more leverage. Meaning that it's X axis value is quite extreme from the other data points. This is marked as High leverage and high influence. So, knowing where the Outliers exist in given data and then how to deal with them can save you from potential pitfall of fitting erroneous mathematical model and providing wrong report to customer.



How to detect Outliers

HyperStudy helps you in identifying Outliers using Box plot (Postprocessing-Distribution- icon on the right).

For more information on calculation please refer to Box Plot help page in HyperStudy. (Hint: Use capping methods. Any value which is out of range of 5th and 95th percentile can be considered as outlier)

How to deal with them

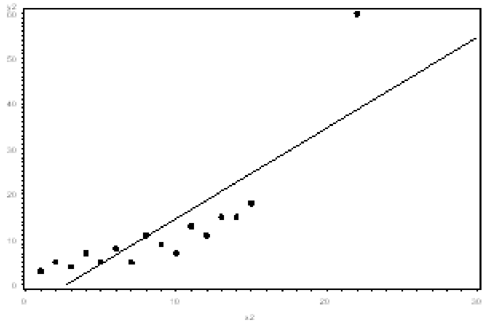
Once you have identified Outliers in your data, next question is how to deal with them. Following points could assist.

1. Does the data make sense: E.g. out of 50 samples which were used to record fatigue life 98 percent show a mean life cycle of 1.5 million and 2 percent show a life cycle of 10,000 cycles. This could be experimental or measurement error. In this case if the customer is positive that the life cycle of a given component cannot be so low then those sample should be removed from data set and documented in final report.
2. In most cases, however, removing data is not so straight forward. Couple of things you can do:

Create a model with and without the outlier(s). If the model remains unaffected (very marginally affected) by the absence of outliers, then you can drop them. Again, this should be documented.

If the model (R^2, Adj. R^2 change drastically) then we could do following:

- One option is to try a transformation. Square root and log transformations both pull in high numbers. This can make assumptions work better if the outlier is a dependent variable and can reduce the impact of a single point if the outlier is an independent variable.
- Another option is changing the model itself. E.g. In the given image, exponential model would do a better job with the outlier intact.



3. Treat separately: If there are significant number of outliers, we should treat them separately in the statistical model. One of the approaches is to treat both groups as two different groups and build individual model for both groups and then combine the output.

In other words:

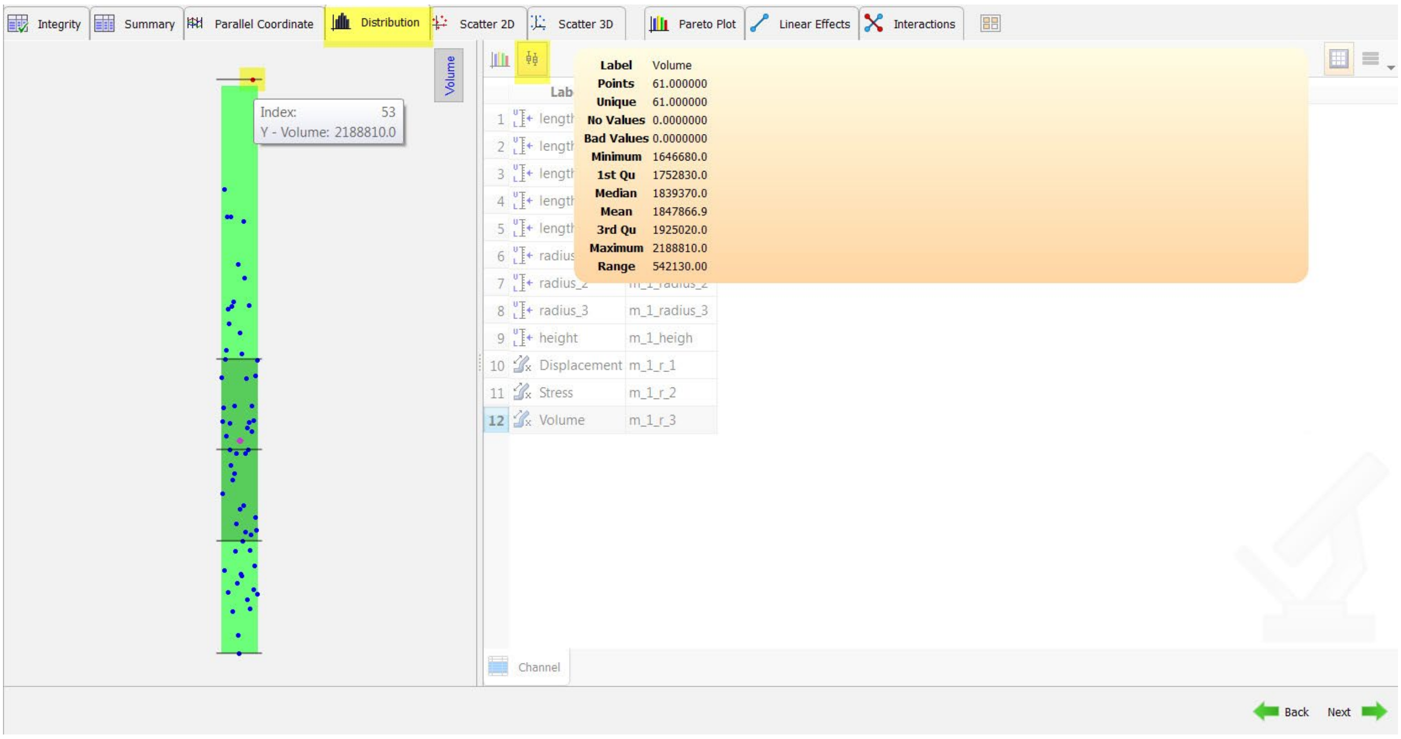
If you look at Postprocessing→Integrity→Health some design may be marked as “Outliers” (see next image).

Integrity | Summary | Parallel Coordinate | Distribution | Scatter 2D | Scatter 3D | Pareto Plot | Linear Effects | Interactions

	Label	Varname	Category	Outliers	LCL	UCL	Min Bound 99%	M
1	length_1	m_1_length_1	Variable	0	-1.4641929	2.8373076	-0.4800000	-0.4
2	length_2	m_1_length_2	Variable	0	-0.7111775	2.7271775	0.0320000	0.0(
3	length_3	m_1_length_3	Variable	0	-1.7257522	1.7564408	-0.9360000	-0.9
4	length_4	m_1_length_4	Variable	0	-1.7310632	1.7322265	-0.9838917	-0.9
5	length_5	m_1_length_5	Variable	0	-1.7545827	1.7361098	-0.9926531	-0.9
6	radius_1	m_1_radius_1	Variable	0	-3.5184052	3.4352989	-1.9521605	-1.9
7	radius_2	m_1_radius_2	Variable	0	-0.8515785	1.4642141	-0.4268479	-0.3
8	radius_3	m_1_radius_3	Variable	0	-1.0365364	1.5221498	-0.4511399	-0.4
9	height	m_1_heigh	Variable	0	-1.6013487	1.7889657	-0.9892714	-0.9
10	Displac...	m_1_r_1	Response	0	0.7484965	1.9110507	0.9318950	0.9(
11	Stress	m_1_r_2	Response	0	104.84605	360.57682	156.01526	162
12	Volume	m_1_r_3	Response	1	1489893.3	2205840.5	1646680.0	166

	Category
1	Health
2	Summary
3	Distribution
4	Quality

To better understand which run (and hence which combination of design variables) is the cause of the outlier, go to Postprocessing→Distribution→Box Plot (see below).

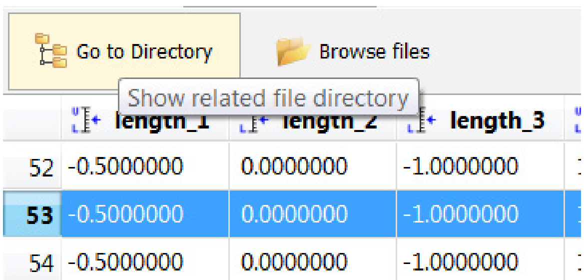


By hovering across the outlier, the run number and the response (here Volume) is shown along with other information such a total number of runs (here 61), minimum and maximum values of the volume.

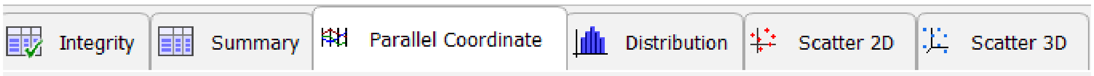
In a linear static run like the one conducted, the outlier refers to a “real” (or reasonable) result, i.e. model run successfully completed. However, in an explicit run (like crash simulation), the outlier may indicate that the run was – let’s phrase it very general – subjected to some issues such as run terminated “accidently”: Maybe convergence was not reached at the time the analysis terminated. In such cases HyperStudy extracts the last available response value which may be far off from all other designs …

Hence, one step you would pursue is to check the model (and results) of the run indicated in the BoxPlot (here run 53). All relevant files are contained in the respective working directory. You can access to it easily by Browsing the folder or Going to the Folder from

Evaluation step or Summary tab of the Post-Processing Step.



6.7.5 Parallel Coordinates (Snake Plot)

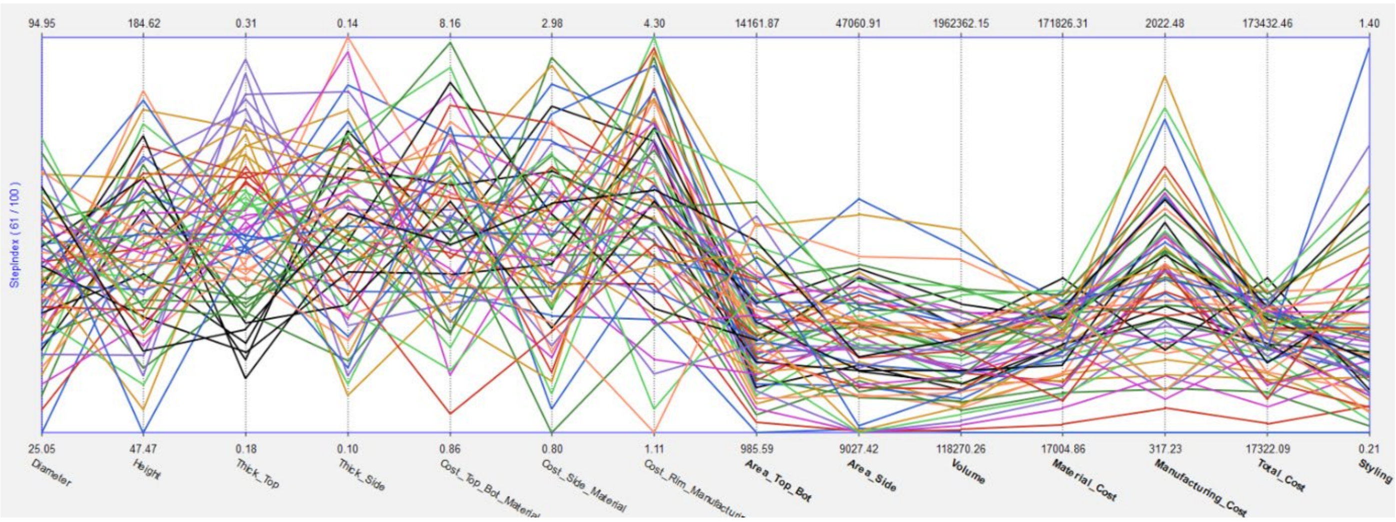


Some characteristics of this plot:

- A vertical, equally spaced parallel line is drawn for each variable and response
- A horizontal line is drawn for each design

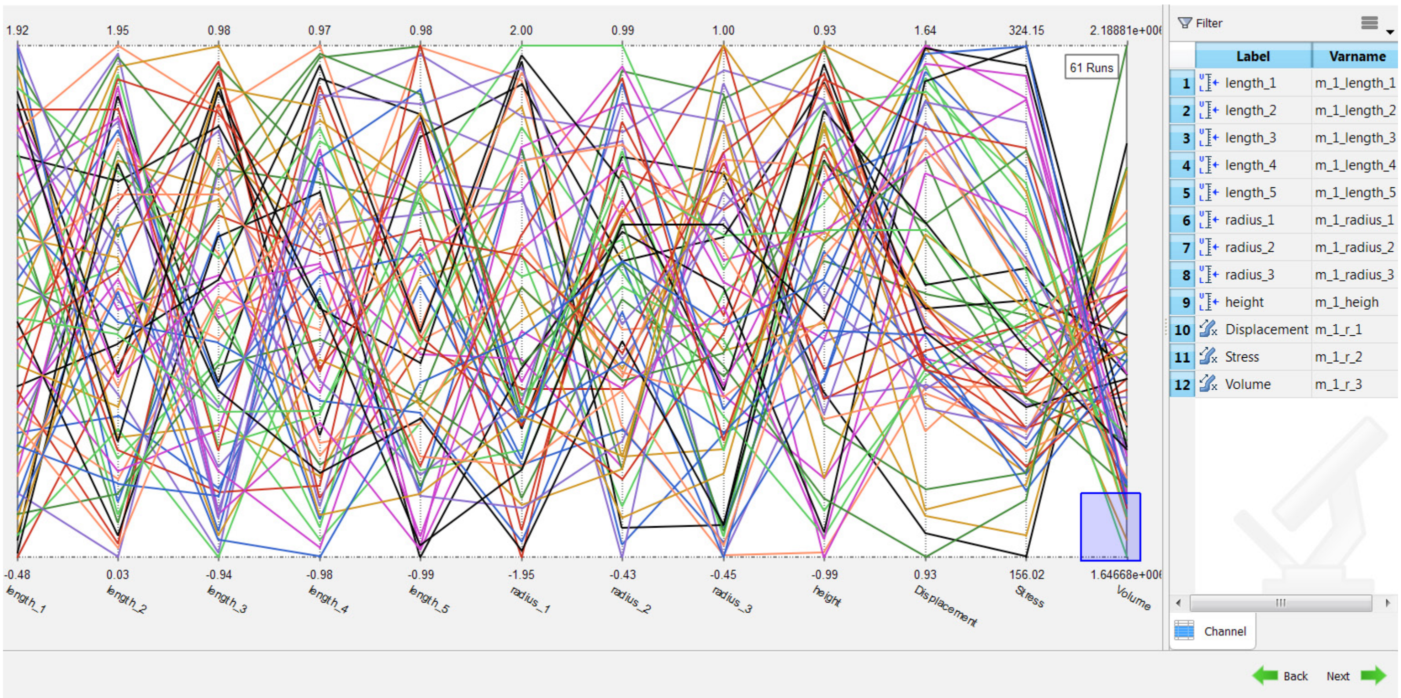
By default, the min and max range for each selected variable/response is displayed at the top and bottom of an axis. Each run in the data is represented as a colored line passing through the axes.

- Helps to identify patterns, clusters, relations
- Data can be filtered to focus on a range
- Order can be changed to reveal relations

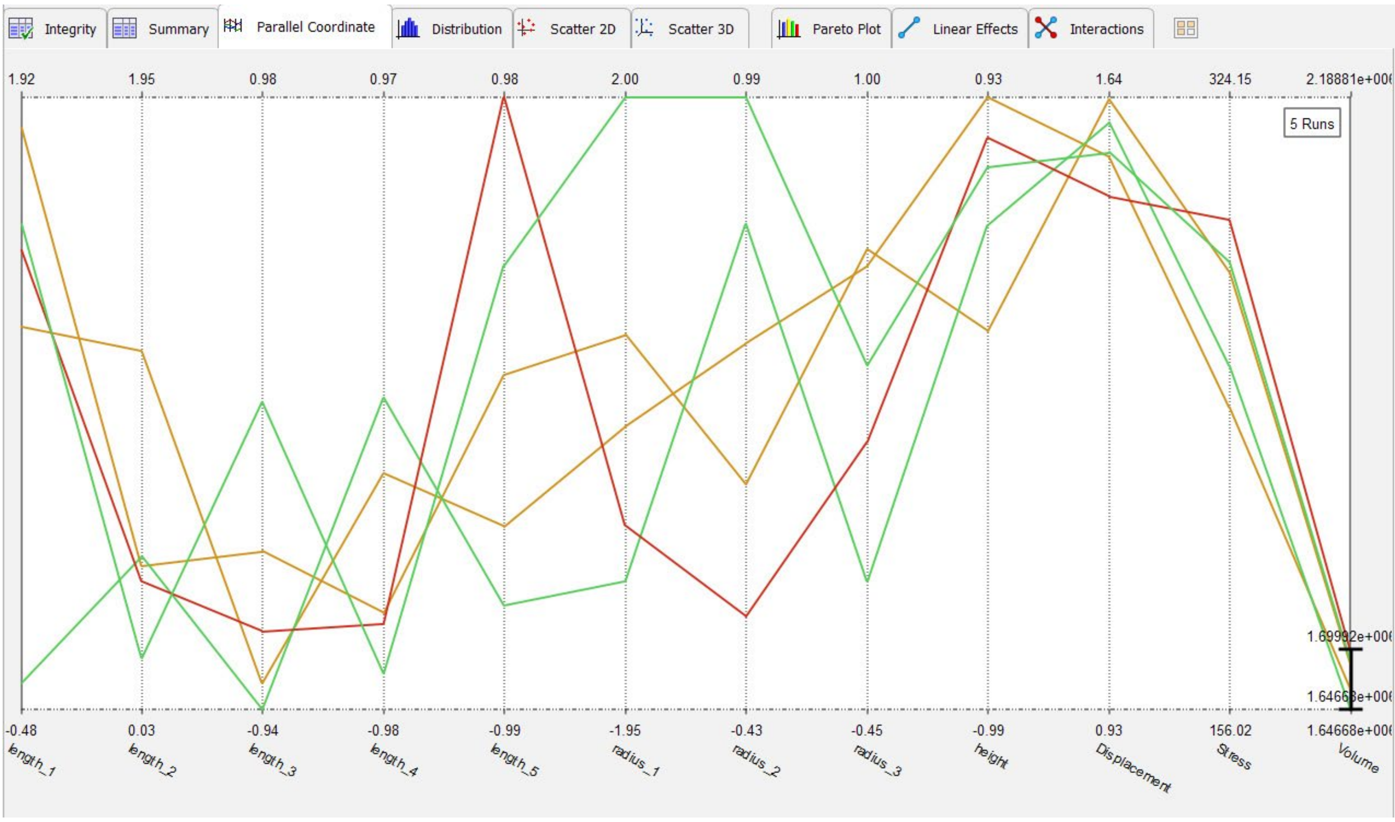


For instance, the “Diameter” (shown in the left) ranges from 25.05 to 94.95, Height (second vertical line) ranges from 47.47 to 184.62 etc. By picking a curve (design of interest) you get another graphical feedback regarding the importance/impact of a particular design variable on this design.

Alternatively, you may look at a particular response, let’s say Volume. If your project aim is to find a design within a given Volume range, span (open) an appropriated window (left mouse) ver the acceptable Volume range. As a result, all model configurations (here 5) which result in the desired volume range (shown in the lower right corner) are depicted. In addition, the graph helps to understand the effects of individual design variables (e.g. height) on the Volume (for instance, height significantly the volume).



Snake plot



Sake plot, local zoom

Overall, the above described visualization methods are applied to better understand the data, e.g. to see trends etc. By using a combination of these methods, you should see effects (or be able to extract information) in the data you wouldn't see by restraining/focusing on a single plot/graph.

6.7.6 Scatter 2D / 3D Plots (Anthill Plots)

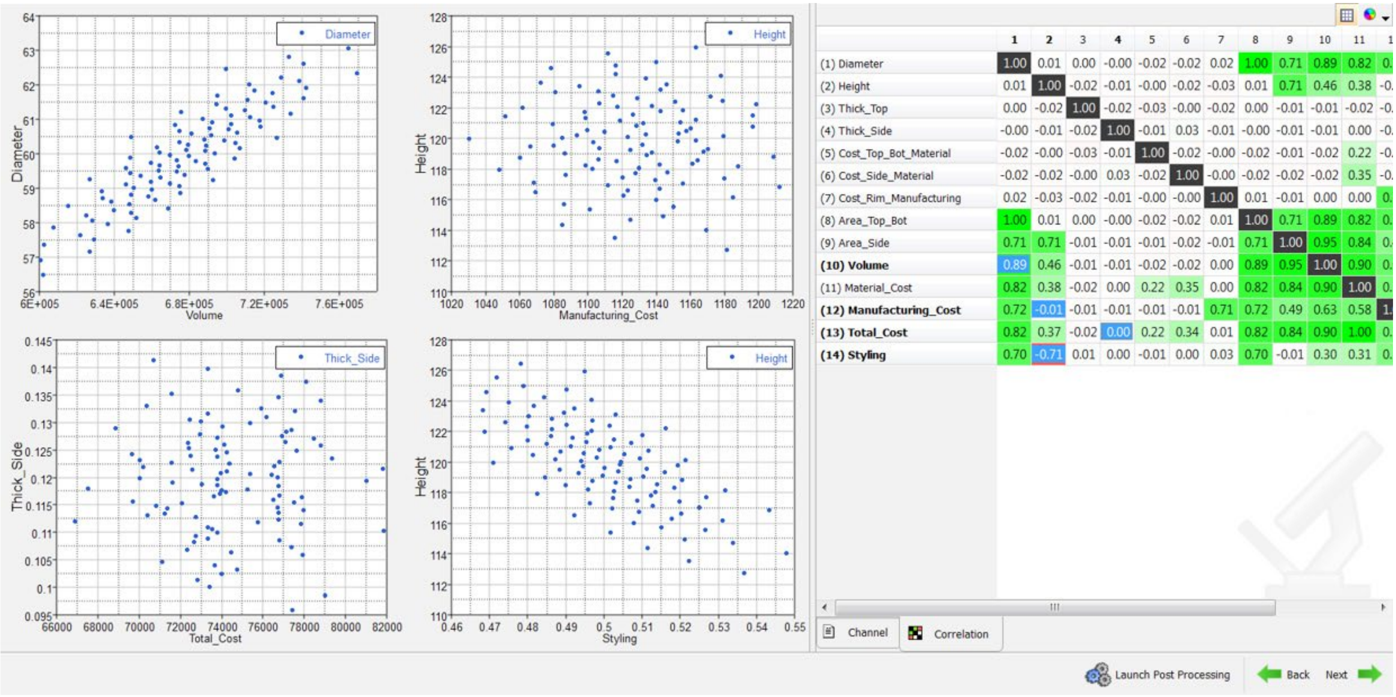
Scatter plots (or anthill plots) are 2D representations of all responses and variables. Each dot stands for one computation (= one line of the table). The X and Y axis can be any column of the data set. The purpose of the scatter plots is to graphically highlight the relationships between a variable and a response or a response and a response.

Some characteristics:

- displays all variables and responses.
- indicates the dependency between the two axes.

Correlation coefficients

- can have a value from -1 to 1
- -1 indicates a strong but negative correlation and 1 indicates a strong and positive correlation.



	1	2	3	4	5	6	7	8	9	10	11	1
(1) Diameter	1.00	0.01	0.00	-0.00	-0.02	-0.02	0.02	1.00	0.71	0.89	0.82	0.
(2) Height	0.01	1.00	-0.02	-0.01	-0.00	-0.02	-0.03	0.01	0.71	0.46	0.38	-0.
(3) Thick_Top	0.00	-0.02	1.00	-0.02	-0.03	-0.00	-0.02	0.00	-0.01	-0.01	-0.02	-0.
(4) Thick_Side	-0.00	-0.01	-0.02	1.00	-0.01	0.03	-0.01	-0.00	-0.01	-0.01	0.00	-0.
(5) Cost_Top_Bot_Material	-0.02	-0.00	-0.03	-0.01	1.00	-0.02	-0.00	-0.02	-0.01	-0.02	0.22	-0.
(6) Cost_Side_Material	-0.02	-0.02	-0.00	0.03	-0.02	1.00	-0.00	-0.02	-0.02	-0.02	0.35	-0.
(7) Cost_Rim_Manufacturing	0.02	-0.03	-0.02	-0.01	-0.00	-0.00	1.00	0.01	-0.01	0.00	0.00	0.
(8) Area_Top_Bot	1.00	0.01	0.00	-0.00	-0.02	-0.02	0.01	1.00	0.71	0.89	0.82	0.
(9) Area_Side	0.71	0.71	-0.01	-0.01	-0.01	-0.02	-0.01	0.71	1.00	0.95	0.84	0.
(10) Volume	0.89	0.46	-0.01	-0.01	-0.02	-0.02	0.00	0.89	0.95	1.00	0.90	0.
(11) Material_Cost	0.82	0.38	-0.02	0.00	0.22	0.35	0.00	0.82	0.84	0.90	1.00	0.
(12) Manufacturing_Cost	0.72	-0.01	-0.01	-0.01	-0.01	-0.01	0.71	0.72	0.49	0.63	0.58	1.
(13) Total_Cost	0.82	0.37	-0.02	0.00	0.22	0.34	0.01	0.82	0.84	0.90	1.00	0.
(14) Styling	0.70	-0.71	0.01	0.00	-0.01	0.00	0.03	0.70	-0.01	0.30	0.31	0.

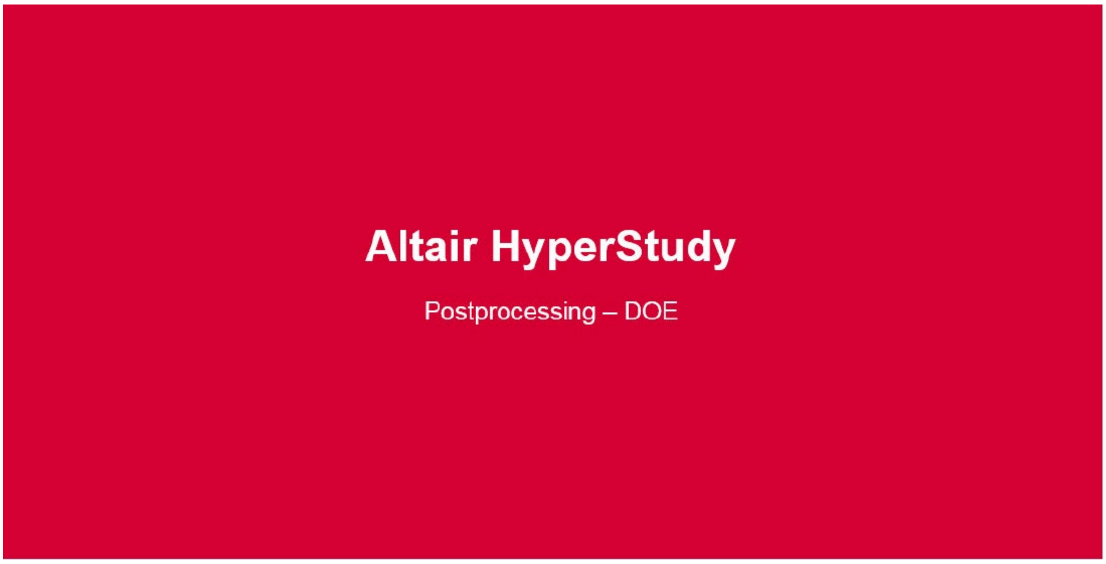
In the left upper insert / chart a 2D scatter plot depicting the effects of Volume and Diameter are shown. Quite obviously, there is a correlation between both design variables. The picture looks very different if we plot Manufacturing costs vs. Height. Here the data are scattered; there is no clear pattern.

Alternatively, you may create a correlation plot as shown on the right side. In here "1" refers to a positive correlation, -1 indicates a negative correlation.

One of the most powerful aspects of a scatter plot, however, is its ability to show nonlinear relationships between variables.

Furthermore, if the data is represented by a mixture model of simple relationships, these relationships will be visually evident as superimposed patterns.

6.8 Postprocessing - DOE Specific



https://altair-2.wistia.com/medias/z61qvrmn49

The DOE specific options are shown on the right side of the image:

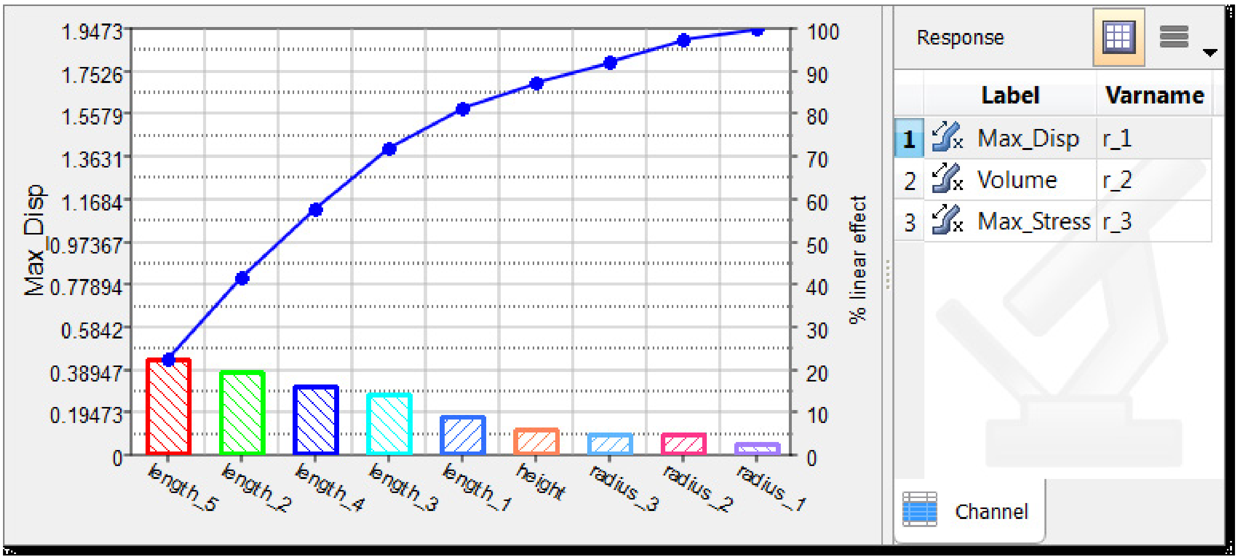
- Pareto Plot
- Linear Effects (Effects table and Linear Effects Plot)
- Interactions (Interactions Table and Interactions Plot)

6.8.1 Pareto Plot

This is a DOE post-processing technique that presents the effects of the variables on a response in a bar chart which ranks the effects from largest to smallest. These simple to understand charts are an effective tool to present the results of a DOE.

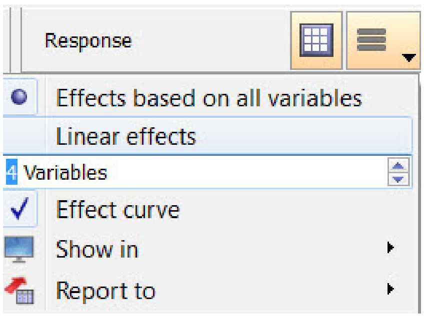
Hashed lines with a positive slope indicate a positive effect (have a close look at the colored “columns” in the image below). If a variable increase, the response will also increase. Hashed lines with a negative slope indicate a negative effect. Increasing the variables lowers the response.

For instance, in the example shown below increasing “length_5” has a negative effect on the Max_Disp and hence would result in smaller displacements. Increasing “length_1” has a positve effect and would result in a larger maximum displacement.



Pareto plot

In order to control the number of displayed variables or to activate the “Effect curve” click on the following icon



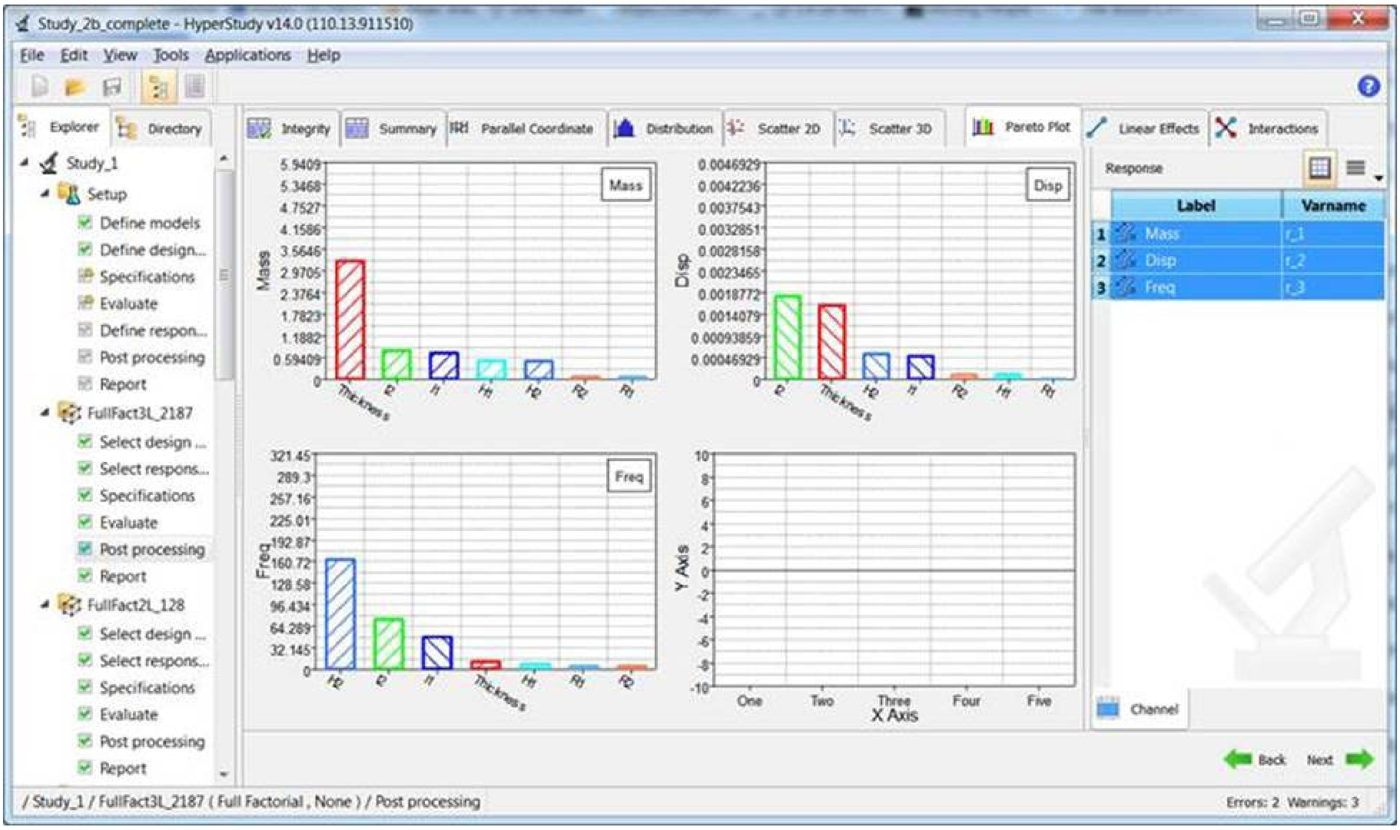
Pareto settings

This panel allows you to control the information being displayed:

- “Effects based on all variables”. When enabled, the effect is calculated using all variables simultaneously.
- “Linear effects”. When enabled, the effect is calculated using each variable independently (same as linear effects)
- “Displayed variable selector”. Controls the number of variables (bars) displayed in the plot. This setting does not change the calculated effects
- “Effect curve”. Displays a line to show the accumulation of total effect

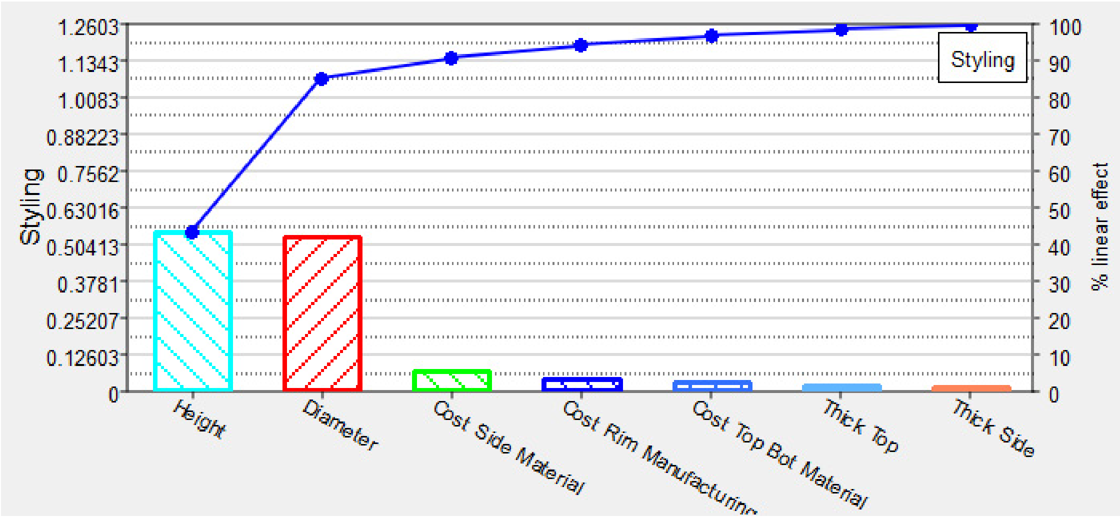
Looking at the plots below, one can quickly see that the red bars are very important to both of the first two responses. Looking at the labels, this means that “thickness” is important for “Mass” and “Disp”. The variable – color combinations are consistent for all

Pareto Plots shown at a given time. This is useful for visualizing the effects of multiple variables on multiple responses at the same time.



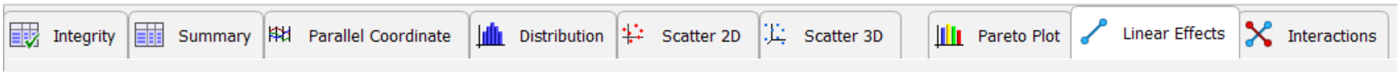
Each bar represents the effect a variable has on the selected response. You can choose to calculate the effect independently (linear effects) or collectively (multivariate). The line, called the Effect Curve, indicates the cumulative effect. Effect curve is what you want to look at if you want to check whether the Pareto Principle holds true for your case. In the example Pareto plot below, 80-20 rule is valid as only a small subset of the design variables has a significant effect on the response, Styling.

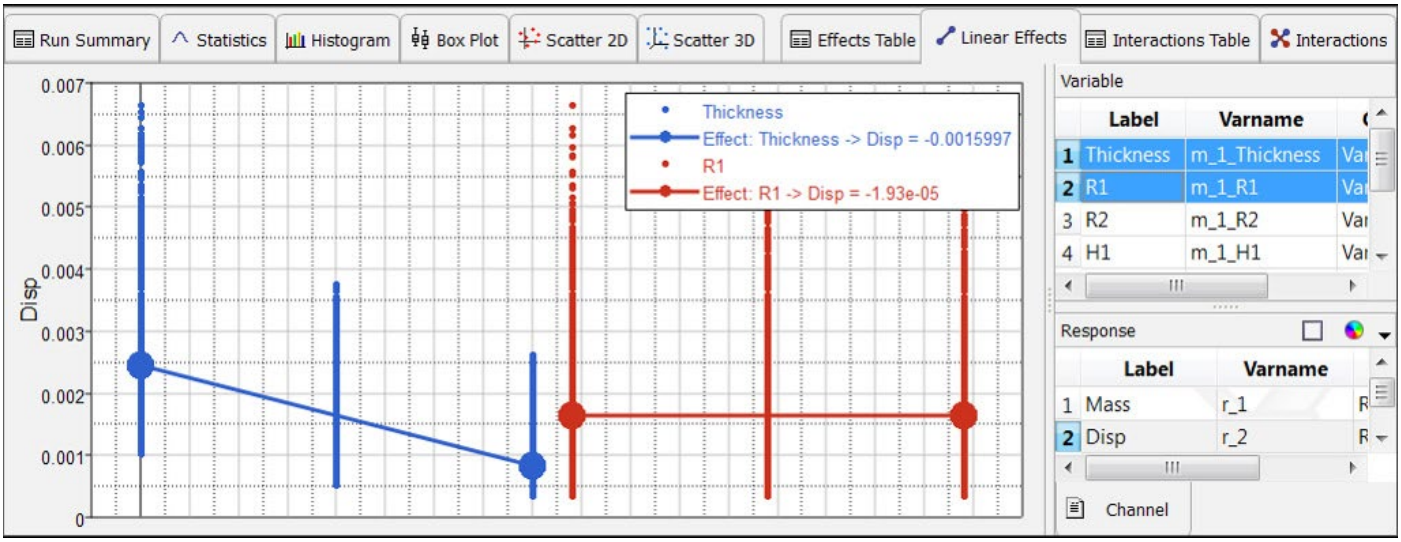
What else do you observe in this example plot (hint: check the hashed line slopes)?



6.8.2 Linear Effects

Linear effects are plotted by drawing a line between the average value of the response when the design variable is at its lower bound and the average value of the response when the design variable is at its upper bound.





As a reminder: A horizontal line implies that this design variable has no impact on the design performance (red curve), whereas the blue curve depicts a negative relationship of the design variable (here Thickness) on the design (for a small value of the design variable Level -1 the response (here "Displacement") is at about 0.0025, whereas at its higher bound Level +1 the Displacement is less than 0.001.

For completeness the definition of Interactions and Effects are repeated here:

6.8.3 Interactions

An interaction is the failure of one variable to produce the same effect on the response at different levels of another variable.

In other words, the strength, or the sign (direction) of an effect is different depending on the value (level) of some other variable(s).

An interaction can be either positive or negative.

In the Interactions tab you can view the effect of a design variable on a response at varying levels of other design variables in an interaction plot or interaction table.

To change the format interactions are displayed, click (Interactions Plot) or (Interactions Table) above the Channel selector.

For the design matrix below, interactions are calculated as:

Effect of X when Y = +1 is (401 - 401) / 2 = 0

Effect of X when Y = -1 is (1601 - 1) / 2 = 800

Interaction of X on Y is then (0 - 800) = -800

Effect of Y when X = +1 is (401 - 1601) / 2 = -600

Effect of Y when X = -1 is (401 - 1) / 2 = 200

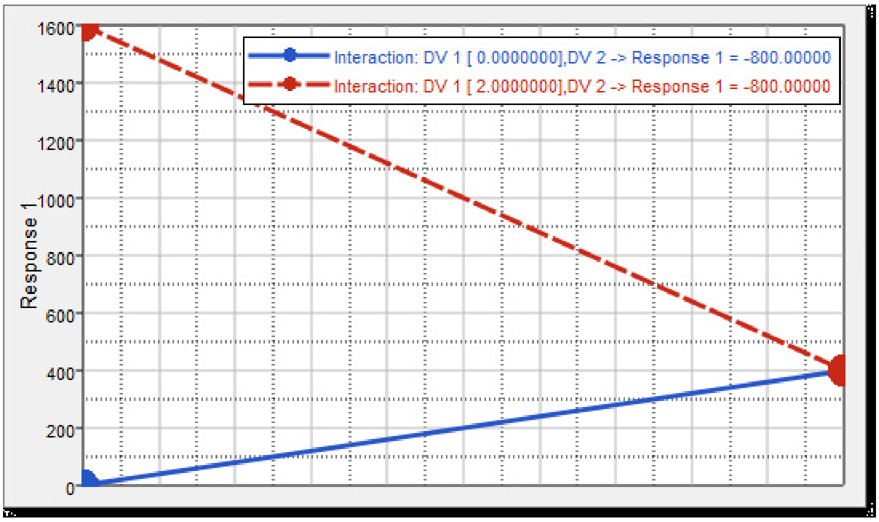
Interaction of X on Y is then (-600 - 200) = -800

Run	X	Y	F(x,y)
1	0	0	1
2	0	2	401
3	2	0	1601
4	2	2	401

Design matrix

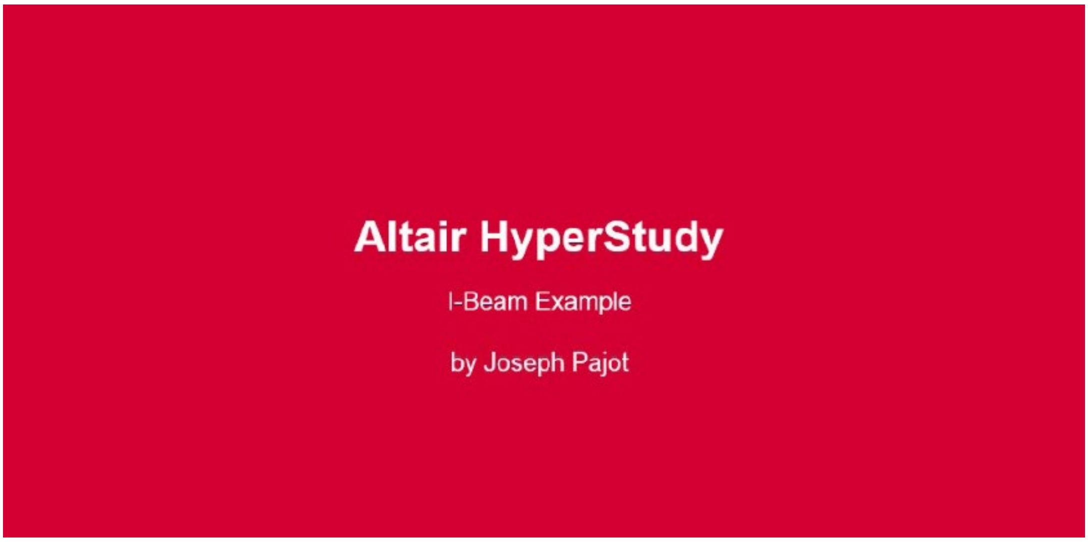
Note that interactions are symmetric; that is:

Interaction XY = effect of (X) on effect of (Y) = effect of (Y) on effect of (X)



Interactions plot

In the I-Beam example recorded by Joseph Pajot the before mentioned steps are discussed again.



https://altair-2.wistia.com/medias/lr6pahlwsp

For a casual blog article summarizing post-processing, have a look at http://innovationintelligence.com/confessions-of-a-dataset/

7 Introductory Examples

In the following hands-on exercise, we'll start with a HyperMesh model. In other words, we are going to start up HyperStudy from within HyperMesh.

Note: Basic HyperMesh skills are required, as we are not going to give an introduction in HyperMesh here – even though we show

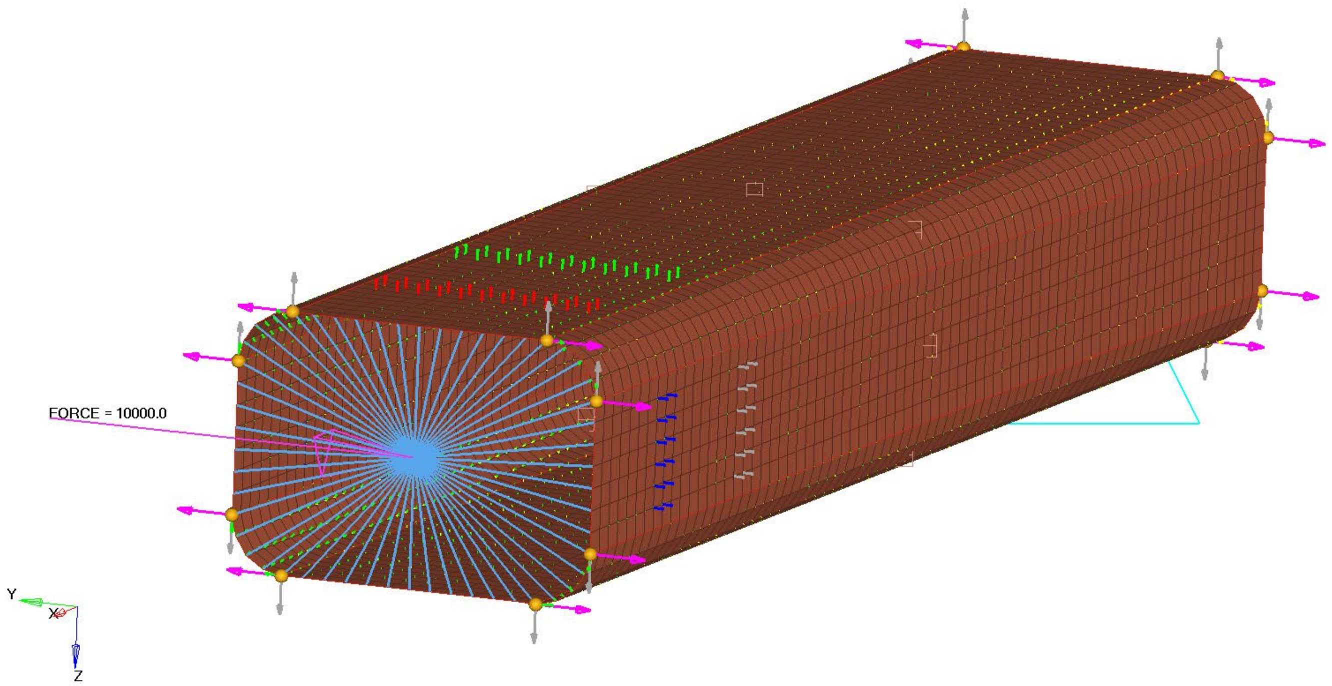
some very basic steps in this first tutorial. In case you need to learn HyperMesh, please attend the free eLearning course: "Learn Pre-Processing with HyperMesh" available in the Learning and Certification Program (http://certification.altairuniversity.com/) or /and the free eBook: Practical Aspects of Finite Element Simulation (http://www.altairuniversity.com/free-ebooks-2/free-ebook-practical-aspects-of-finite-element-simulation-a-study-guide/)

In the successional exercises (further below) we will learn how to:

- Parameterize a solver deck in HyperStudy
- Work with a HyperMesh and HyperMorph Model in HyperStudy
- Work with an Internal Math model in HyperStudy

7.1 Study Setup for a HyperMesh Model with TH, Size and Shape Variables

This exercise outlines the procedure for selecting design variables in HyperMesh and adding them to a study. A HyperMesh database that includes a beam with thickness and shape variables is loaded into HyperMesh. HyperStudy is then launched from HyperMesh and the design variables are selected for HyperStudy.

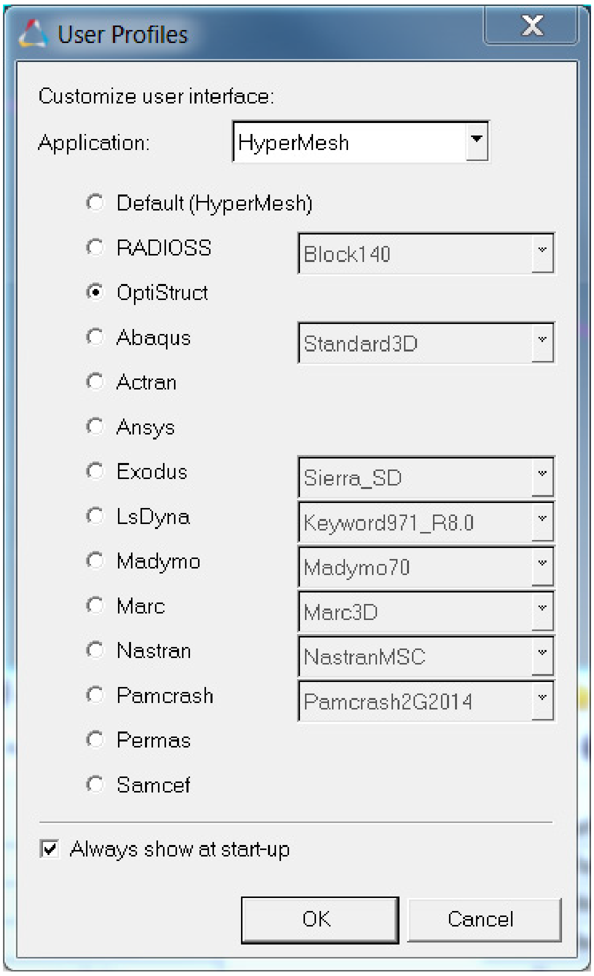


The "beam_shape.hm" model

The file for this exercise is named: beam_shape.hm (see the zipped model files which come along with this eBook). Please copy this file to your working directory first.

7.1.1 Load the File "beam_shape.hm" into HyperMesh Desktop

While the HyperMesh desktop application starts, select the User Profile "OptiStruct"

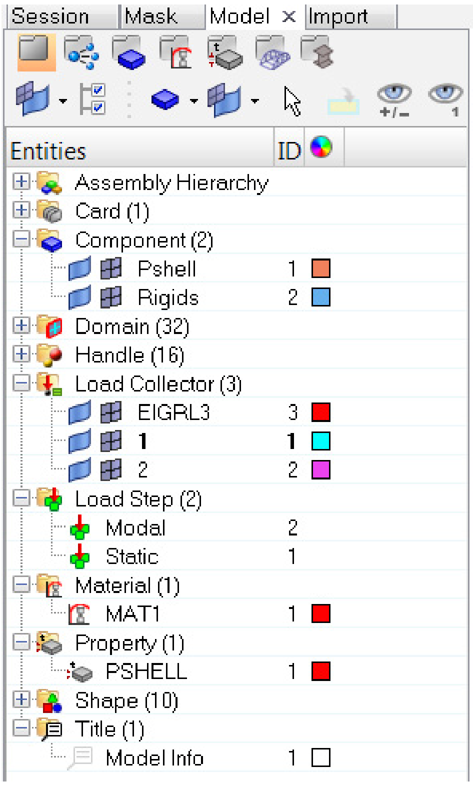


User Profile

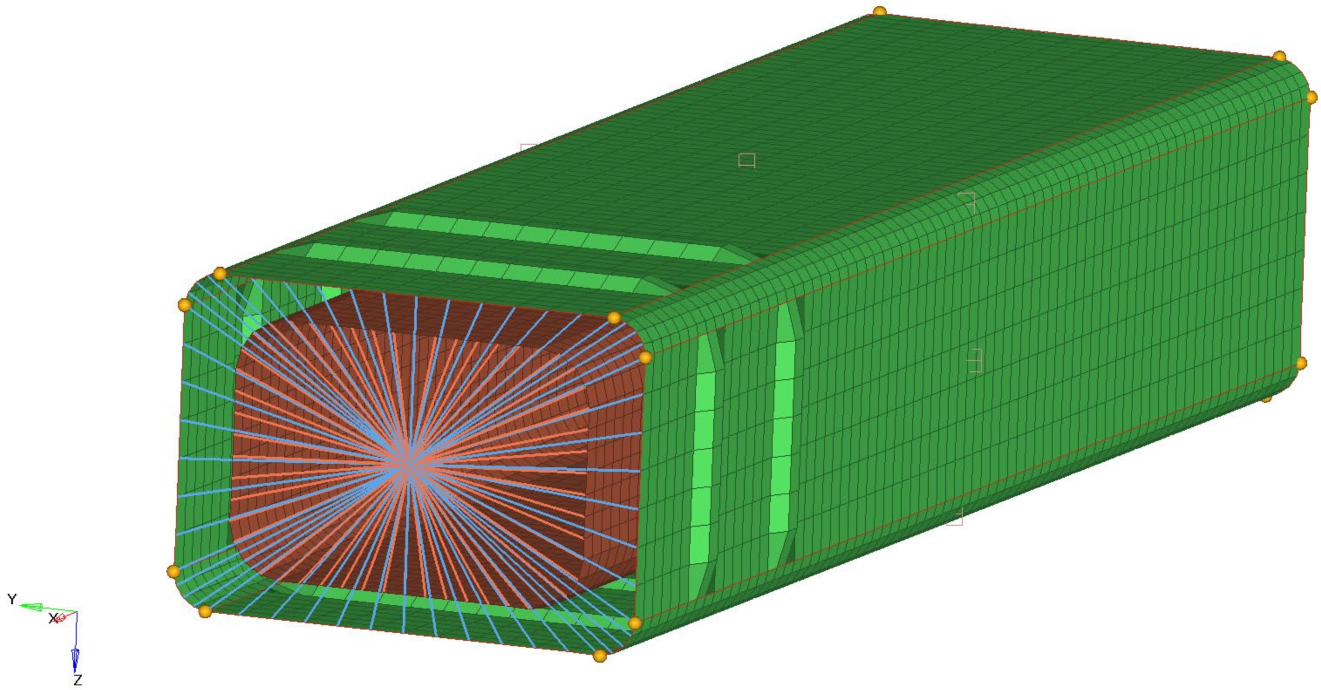
7.1.2 Reviewing the Model

As a common working practice review the given model first.

The loaded HM model already contains all relevant information such as loads, load steps, material, property and most importantly "shape definitions" (Note: the shapes have been created by using the Morphing technology available in HyperMesh).



For visualization purposes all previously defined shapes are applied to the model. In the image below, the brown colored elements represent the base model (without applied shapes), whereas the green colored elements represent the model with applied shapes (i.e. multiplier = 1).



All shapes applied, base model in brown color

Relevant Model Information

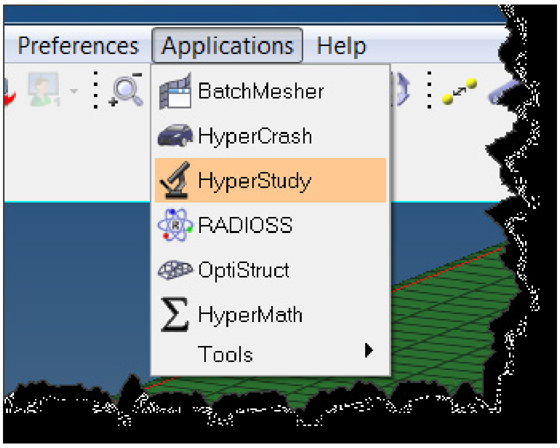
Variables:

R1, R2, H1, H2, I1, I2, 4 Beads along Y and Z, Shell Thickness Define responses:

Mass, Y-Displacement at node 19021, 1st Frequency

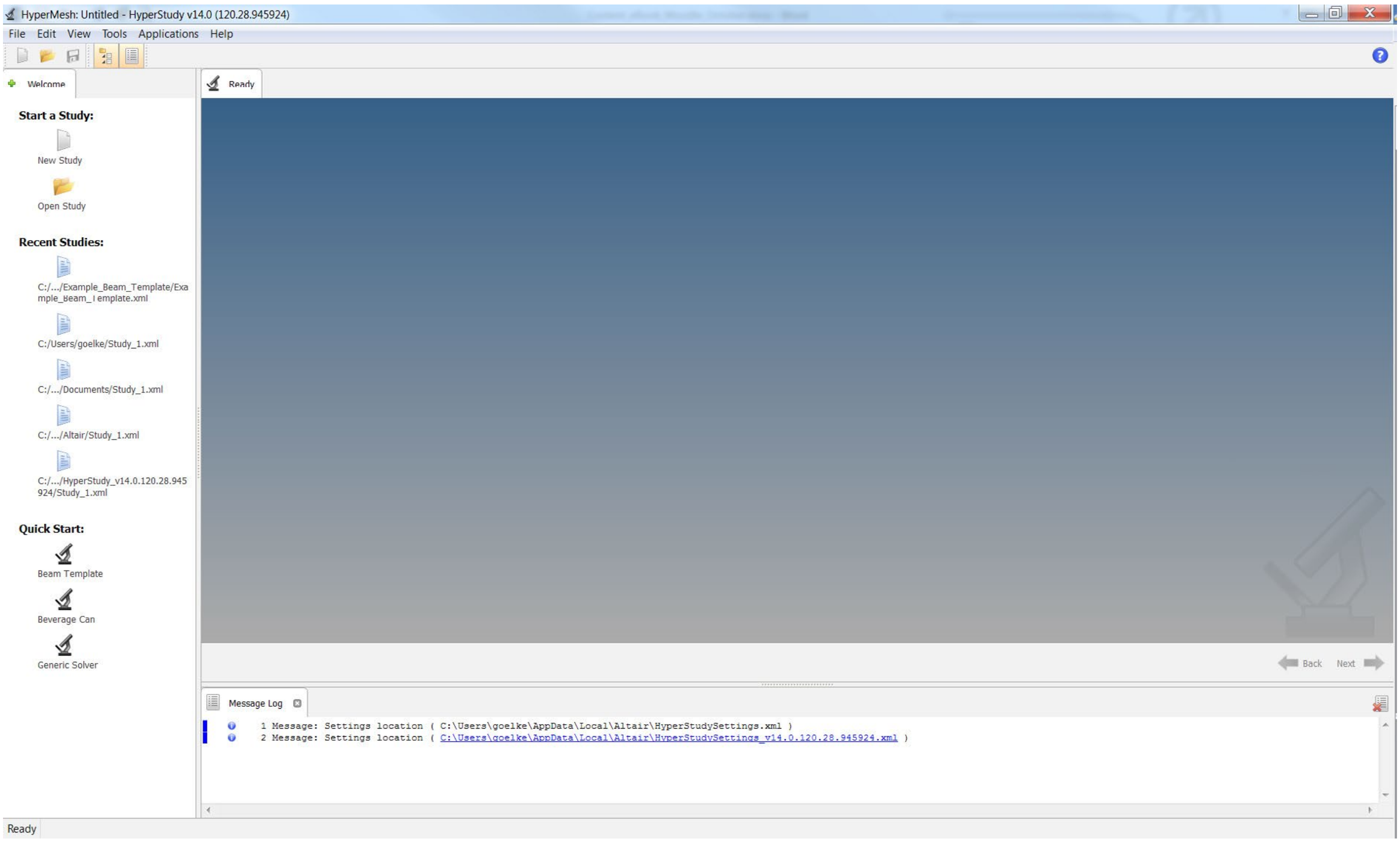
7.1.3 Start HyperStudy and Create a New Study

From the pull-down menu, select Applications → HyperStudy.

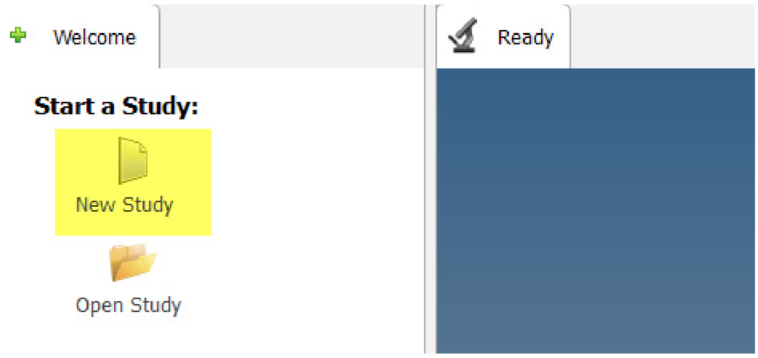


This launches a Message Log and the HyperStudy application and enables the link between HyperMesh Desktop and HyperStudy for reading model variables and enabling model file output for the study.

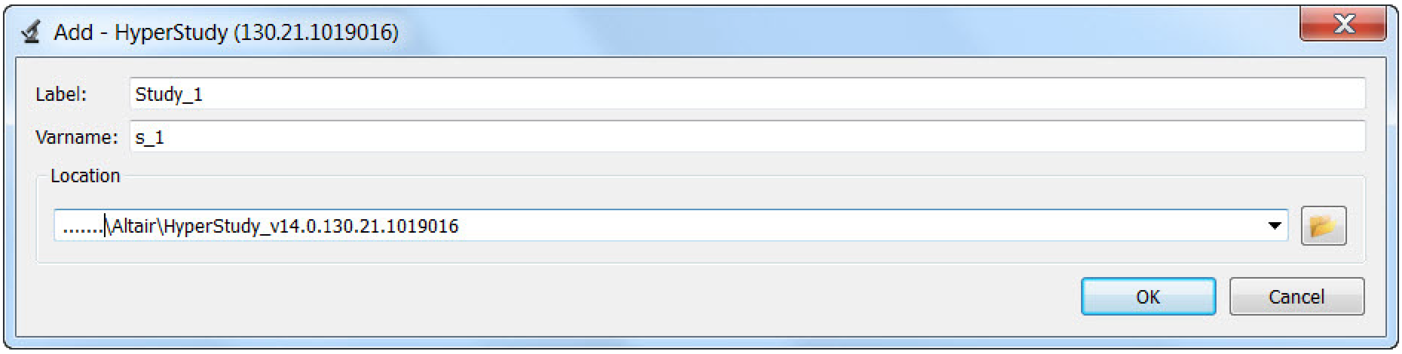
Note: Since the design variables are interactively updated within HyperMesh Desktop, the application must remain open throughout the study, and the new input file for the next run is written out from HyperMesh Desktop.



In HyperStudy, click on "New Study" to create a new study based on the model.

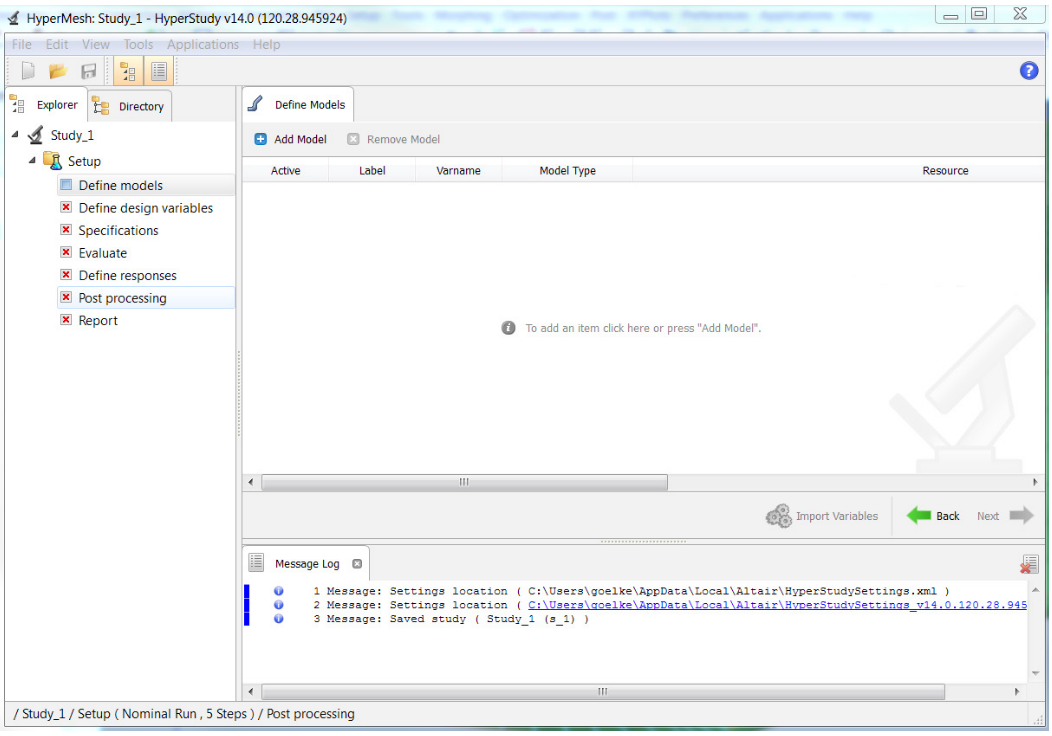


In the HyperStudy–Add window (see image), click OK to use the default labels for the study label and variable.



Note: User interface shown above refers to HyperStudy 14.130

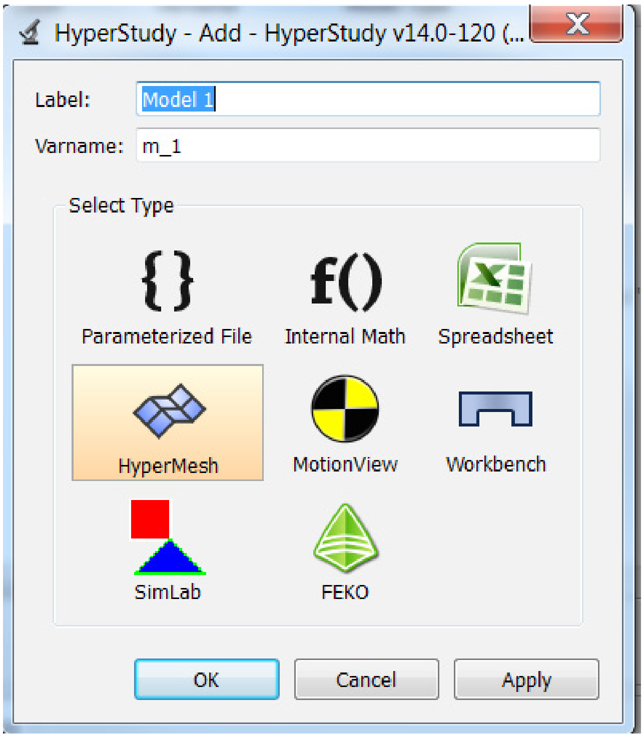
After hitting the “OK” button the following window will be prompted:



We are now in the Study_1 (label used before) Setup.

Clicking on Add Model…

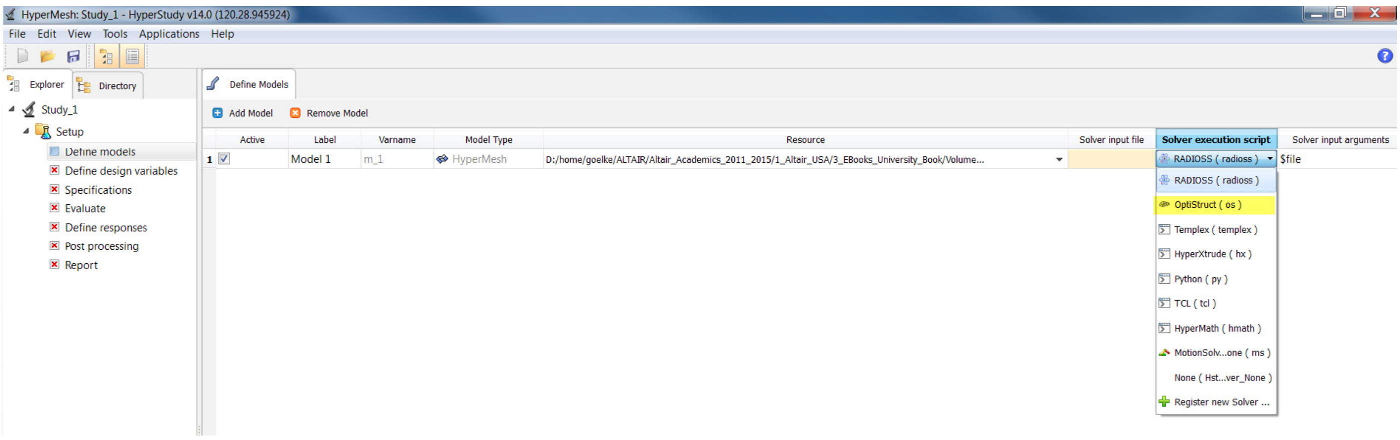
opens up the HyperStudy - Add dialog box.



This dialog box allows you to assign a name label to this model and change the variable type as well as specifying the application origin of the model.

Since the source file is a HyperMesh model, we need to select the HyperMesh icon/symbol. We also accept the default values and add this new model to the study.

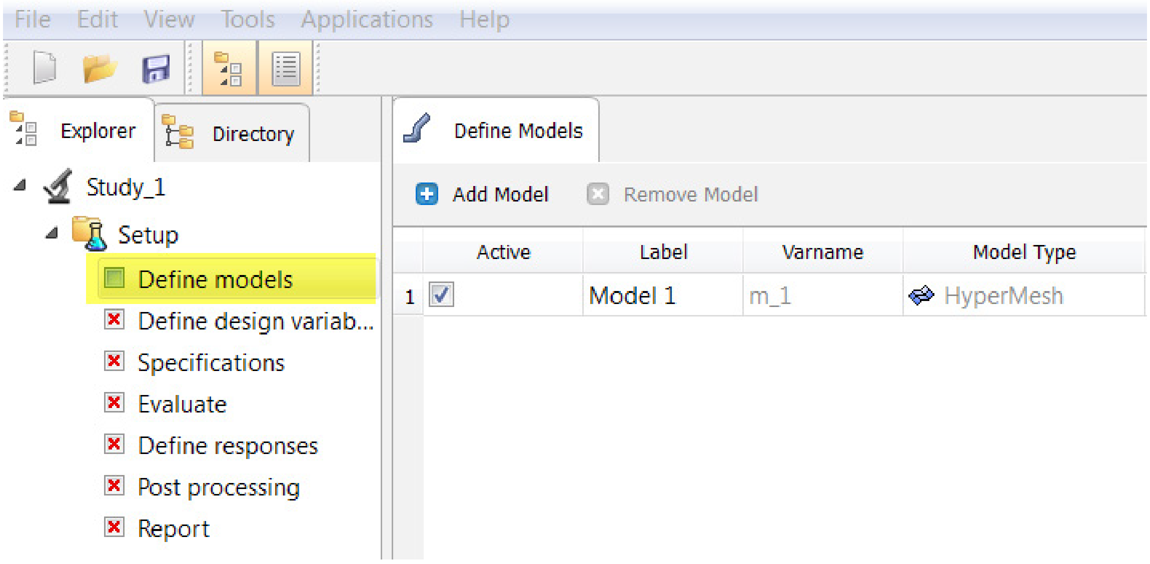
Ensure that the “Solver execution script” drop-down selector is set to OptiStruct (os).



Ensure that the “Solver input arguments” is set to $file. In the “Solver input file” field, enter “beam.fem” (Note: this file doesn’t exist yet. The file name will be assigned to all the different model variants created by HyperStudy).

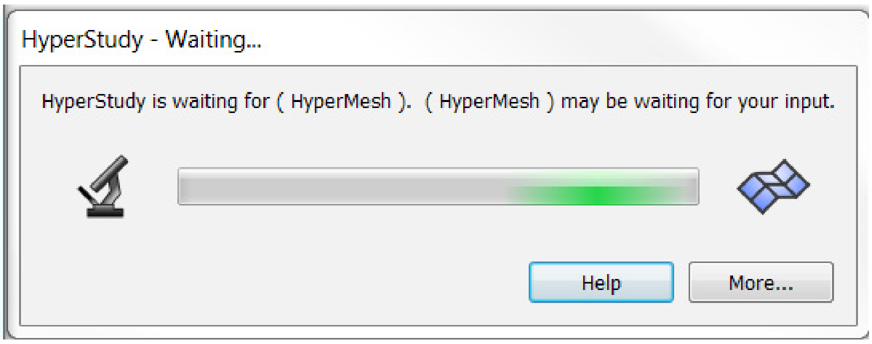
Solver input file	Solver execution script	Solver input arguments
beam.fem	OptiStruct (os)	$file

Also notice that the “Define models” step is not completed yet – otherwise you would see a green check mark.

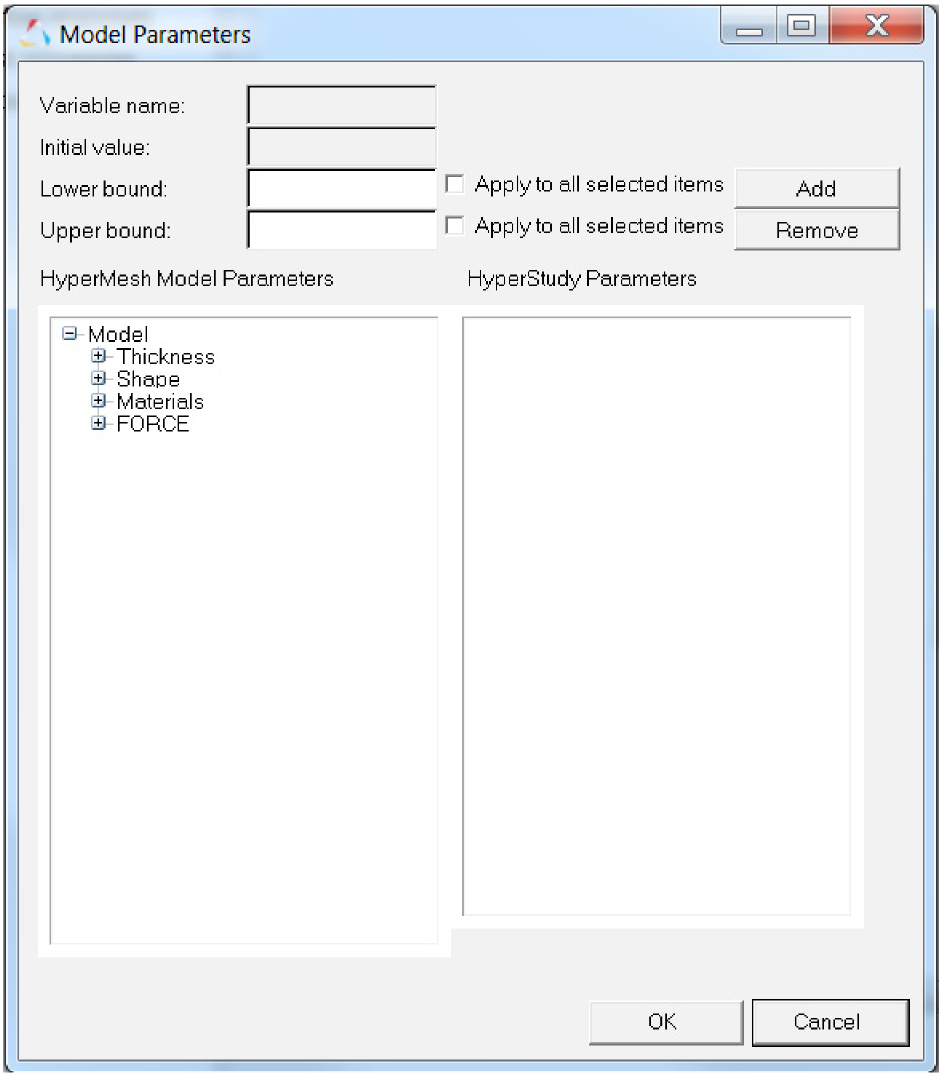


Next, we define, actually import variables (located at the bottom of the Work Area) to bring up the Model Parameters dialog box and to add OptiStruct Model Parameters in HyperStudy.

A Waiting… dialog box will appear to connect HyperStudy with HyperMesh Desktop application (and waits for your input)



Proceed with design variables selection from the “Model Parameters” dialog box.

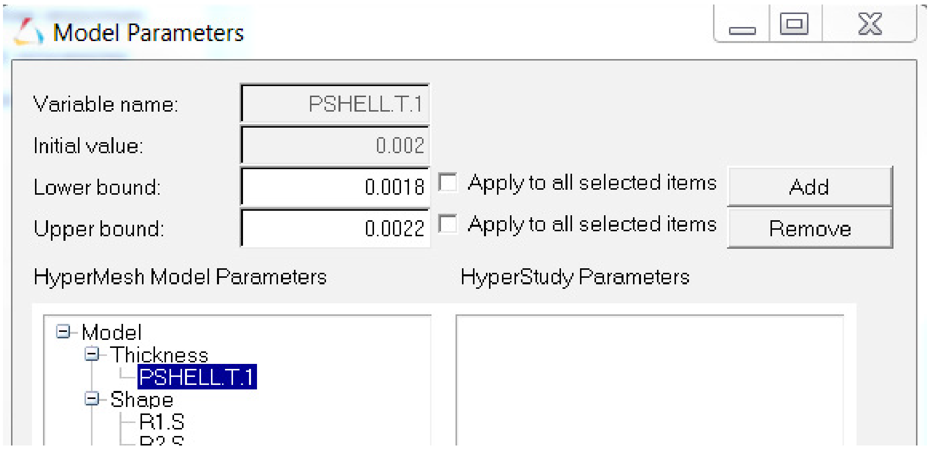


The exciting “thing” is that HyperStudy automatically knows about available model parameters, i.e. Thickness, Shape(s), Materials, and loads (FORCE). This makes the next steps rather easy …

7.1.4 Add Model Parameters

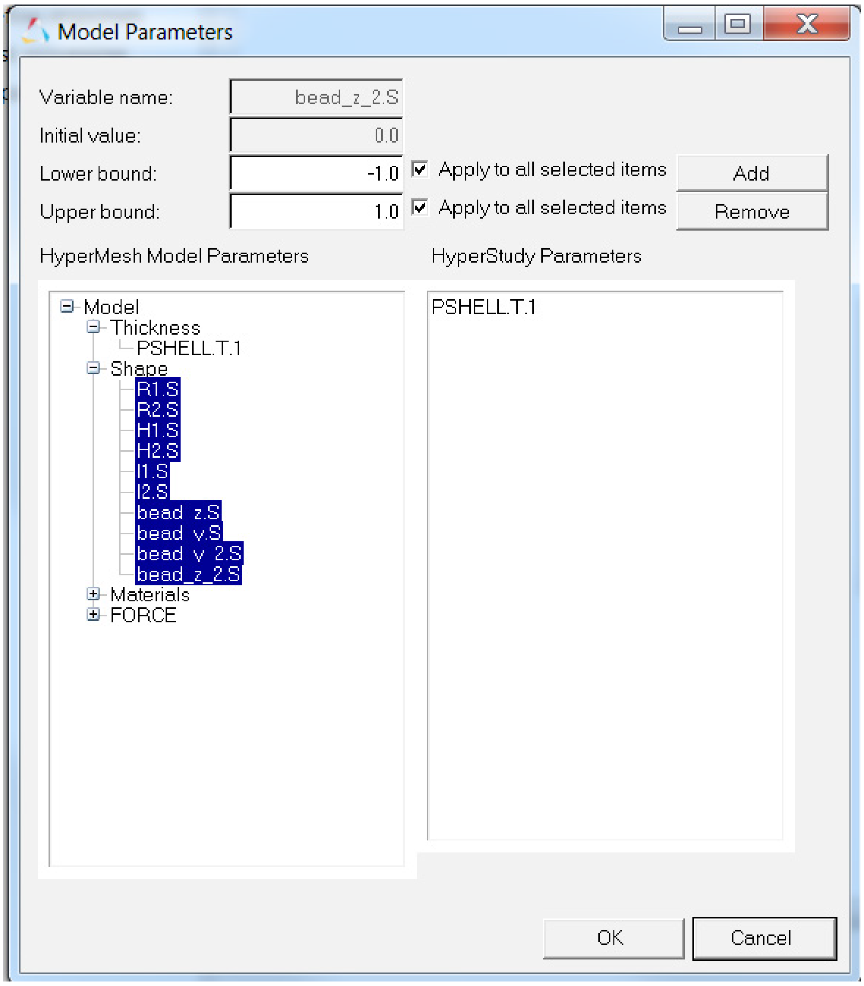
Expand the HyperMesh Model Parameters for “Thickness” and “Shape”; Add all of the variables (Thickness and Shape) shown in the picture below to the HyperStudy Parameters list as well. Click OK to exit the Model Parameters dialog box.

If you select Thickness: PSHELL.T.1, the respective values (initial, lower and upper bound) are displayed. Of course, lower and upper bounds can be customized (even later if needed).



Lower and upper bounds of shell thickness are automatically considered to be +/- 10 % of the initial value. Please note, that this will be discussed in some detail in the Design Variable step of the main user interface.

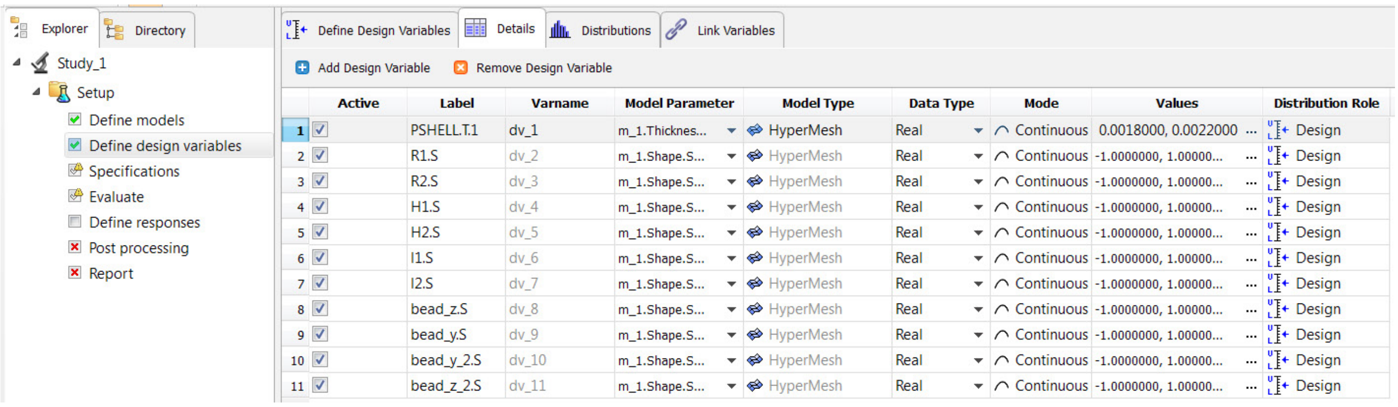
Don't forget to confirm your selection/setting with "Add". Only then the HyperMesh Model Parameters will be added/listed under HyperStudy Parameters.



Note: The Model Parameters window default width may not be wide enough to show the “Add” and “Remove” buttons that are on the top right side of the window. If so, please enlarge the window.

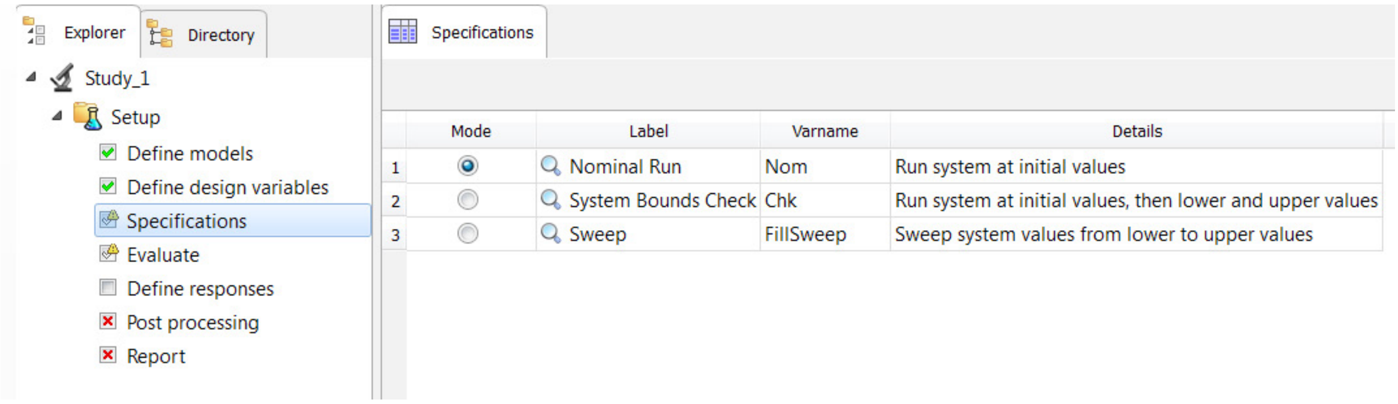
Click “Next” (lower right corner of the screen) to advance to the “Define design variables” step and review the variable properties such as lower and upper bounds, initial values, etc.

The image below depicts information shown in the “Details” tab.



Design variables imported to HyperStudy from the HyperMesh model

Click “Next” to advance to the “Specifications” step to run the Nominal Run.



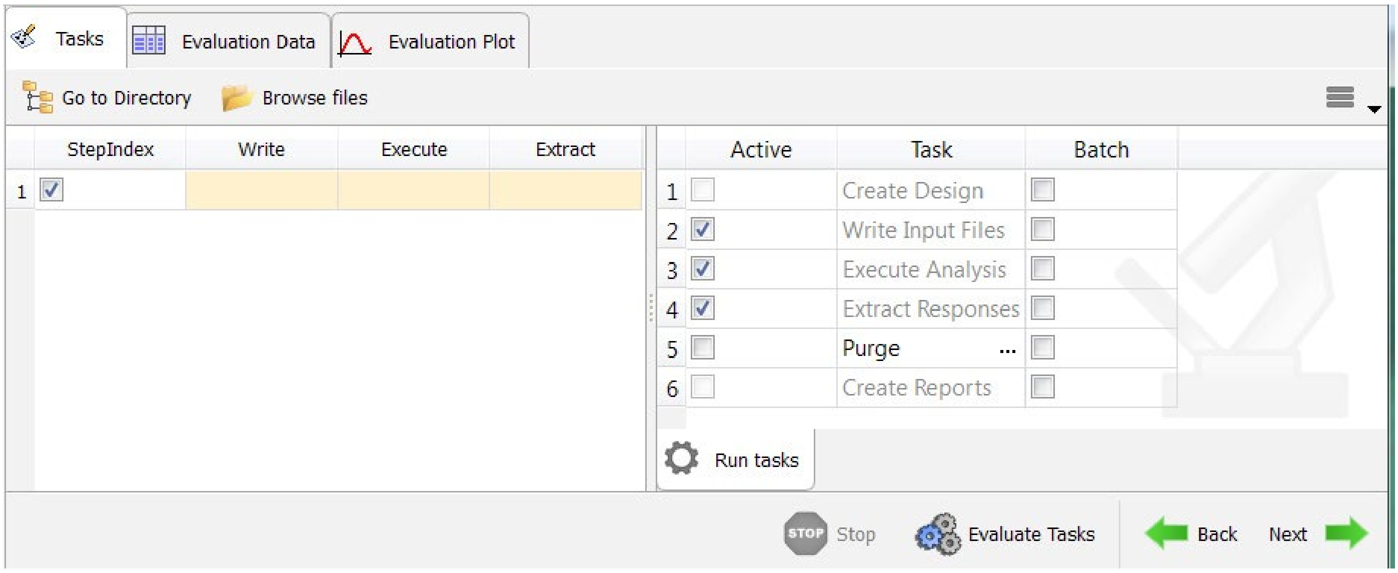
As mentioned under Details, the Nominal Run, runs the model using initial values (e.g. no shapes applied).

7.1.5 Submit the Nominal Run

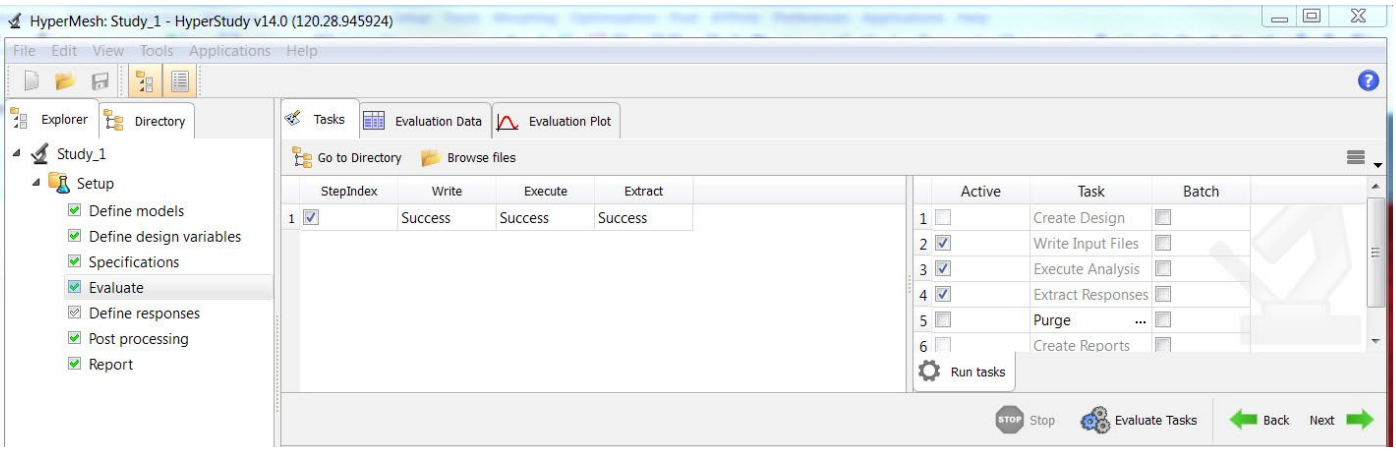
Click on Apply 

to accept the Nominal Run option and then click Next.

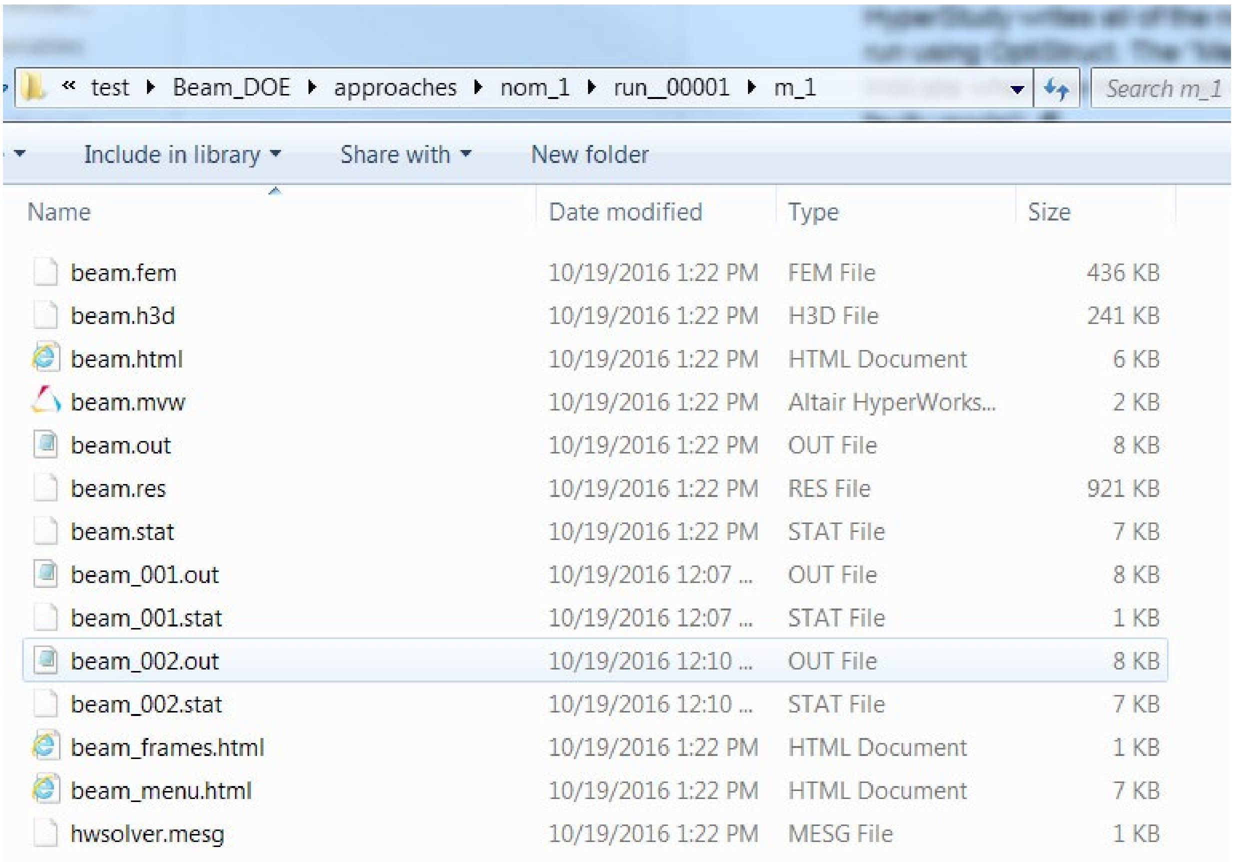
Make sure Write, Execute and Extract have their flag “on”, to write and execute the nominal run for this study.



Click on “Evaluate Tasks” to start up the Nominal Run.

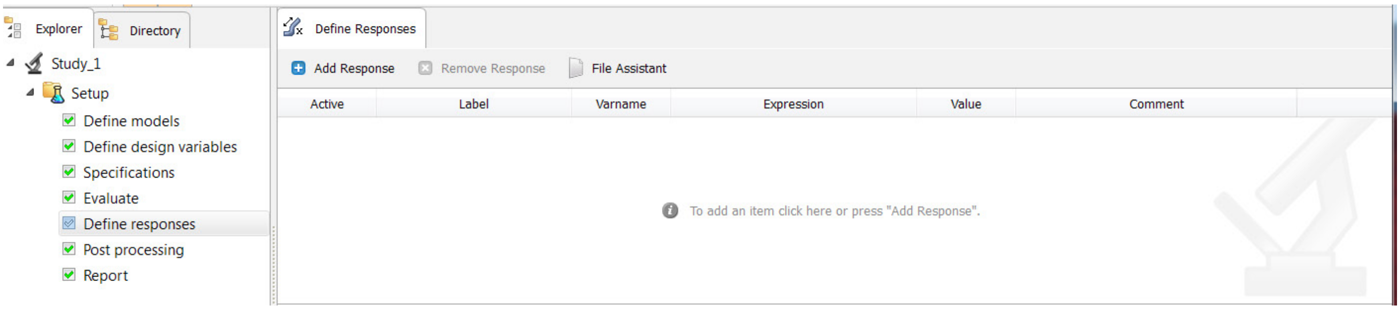


HyperStudy writes all of the necessary input files to the study directory and executes the run using OptiStruct (mind that the files are named beam.* as defined earlier).



The “Messages” window at the bottom of the user interface will indicate when the model has completed execution (or if an error occurred because of a faulty model).

Once the run has completed, click “Next” to proceed to the “Define responses” step.

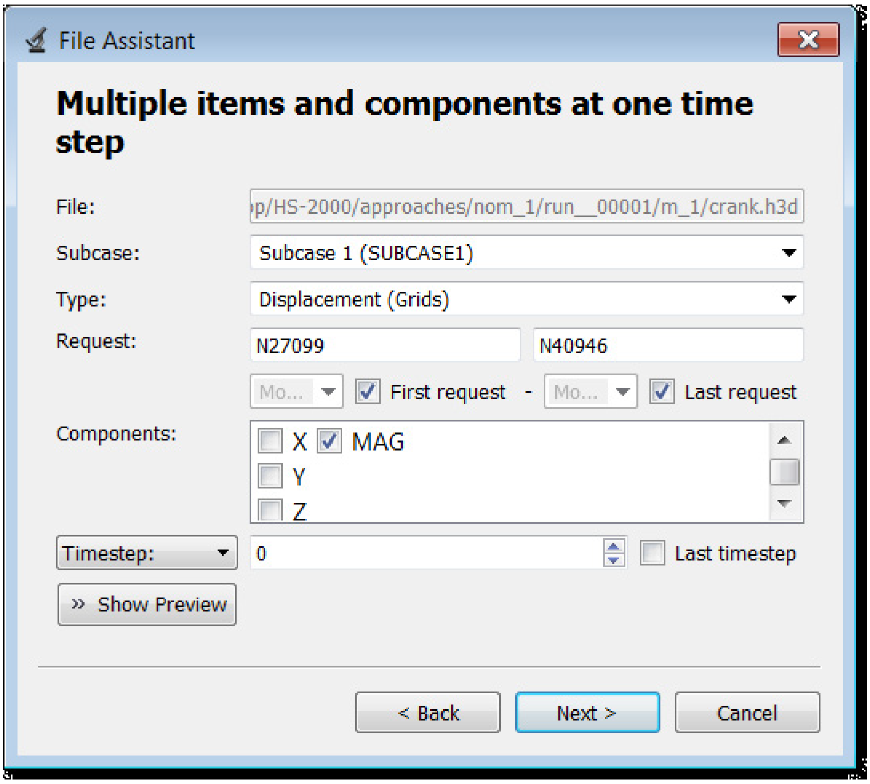


7.1.6 Create Responses and Compare the Response Values

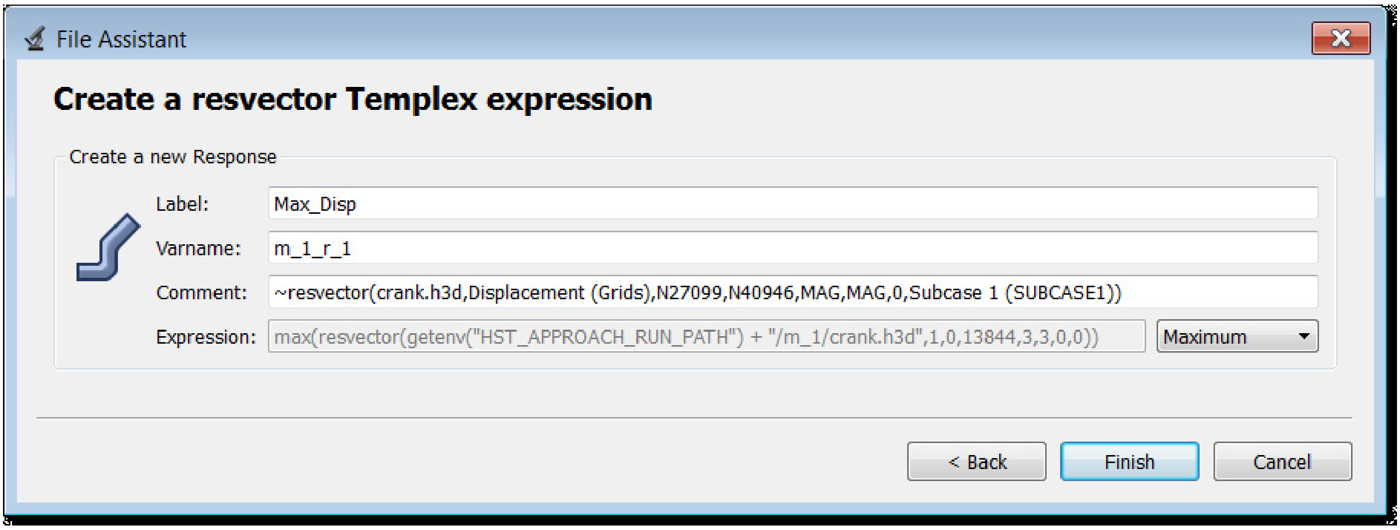
In this step you will create three output responses: Max_Disp, Max_Stress, and Volume. For this purpose you will use the File Assistant. File Assistant guides you through the response setup and it uses the Response Expression Builder in the background. To see how the same responses are defined using the Response Expression Builder, please see Appendix.

Create the Max_Disp output response.

- From the Directory, drag-and-drop the crank.h3d file, located in approaches/nom_1/run_00001/m_1, into the work area.
- In the File Assistant dialog, set the Reading technology to Altair® HyperWorks® (Hyper3D Reader) and click Next.
- Select Multiple items at multiple time steps (readsim), then click Next.
- Define the following options, then click Next.
- Set Subcase to Subcase 1 (SUBCASE1).
- Set Type to Displacement (Grids).
- Set Request (first - last) to N27099 - N40946.
- Set Component to MAG.



- Label the output response Max_Disp.
- Set Expression to Maximum.



- Click Finish. The Max_Disp output response is added to the work area.

Create the Max_Stress output response

- From the Directory, drag-and-drop the crank.h3d file, located in approaches/nom_1/run_00001/m_1, into the work area.
- In the File Assistant dialog, set the Reading technology to Altair® HyperWorks® (Hyper3D Reader) and click Next.
- Select Multiple items at one time step (readsim), then click Next.
- Define the following options, then click Next.
    - Set Subcase to Subcase 1 (SUBCASE1).
    - Set Type to Element Stresses (3D).
    - Set Request (first-last) to E38257 - E94809.
    - Set Component to vonMises (2D & 3D).
- Label the output response Max_Stress.
- Set Expression to Maximum.
- Click Finish. The Max_Stress output response is added to the work area.

Create the Volume output response.

- From the Directory, drag-and-drop the crank.out file, located in approaches/nom_1/run_00001/m_1, into the work area.
- In the File Assistant dialog, set the Reading technology to Altair® HyperWorks® (osmass.tpl) and click Next.
- Select Single item in a time series, then click Next.
- Define the following options, then click Next.
    - Set Type to OptiStruct Analysis.
    - Set Request to Out File.

    - Set Component to Volume.
- Label the output response Volume
- Set Expression to First Element.
- Click Finish. The Volume output response is added to the work area.

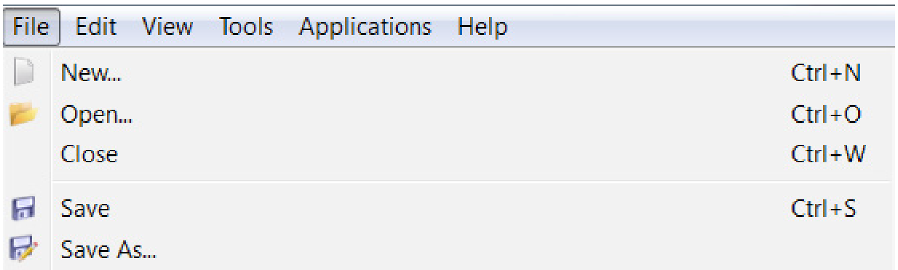
Click Evaluate Expressions to extract the output response values of each expression. The values of the output responses are presented in the table below:

	Active	Label	Expression	Value
1	☑	Max_Disp	max(readsim(getenv("HST_APPROACH_RUN... ...	1.4108263
2	☑	Max_Stress	max(readsim(getenv("HST_APPROACH_RUN... ...	195.29431
3	☑	Volume	m_2_v_1[0] ...	1766760.0

Finally, we can save the Study

7.1.7 Save the Study

In the File menu, select Save / Save As …

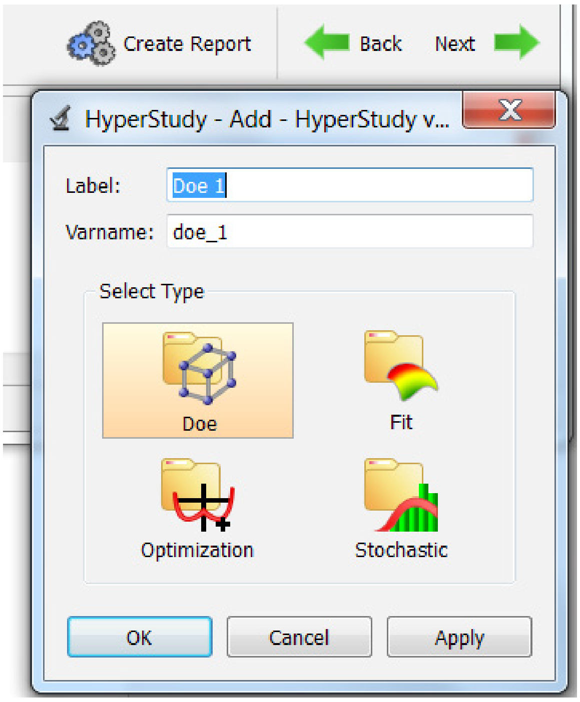


If this study has not yet been saved, a "Save As …" dialog box will pop up requesting a name for the Study archive. The study is then saved as a *.hstx file

This completes the general model / study set up (so far nothing about a DOE was specified). To proceed click on "Next" in the lower right corner of the GUI. You then will be asked to

"Add Approach"

This opens up another window which allows you to proceed with a DOE, FIT, Optimization or Stochastic study.



7.2 Study Setup of a Parameterized File Model for Size Variables

In the previous example the design variables (i.e. Shapes, Material etc.) were directly imported to HyperStudy (to recall: HyperStudy was started from within HyperMesh).

In the following example the procedure for parameterizing the shell element thickness (design variable) using the HyperStudy Editor is shown.

HyperStudy Editor inserts TEMPLEX statements to parameterize the solver input deck with respect to user commands. In HyperStudy Editor, user points to input file, highlights the values that needs to parametrize to be design variables. The editor than inserts templex statements to create the "parametrized input deck" to be used as a resource file for HyperStudy.

This kind of process gives you a great deal of flexibility, also because it is applicable to any ASCII input file, including Altair and non-Altair solvers such as in-house codes.

In this exercise, an OptiStruct beam model is loaded into HyperStudy Editor. The study set-up is shown and a nominal run is performed.

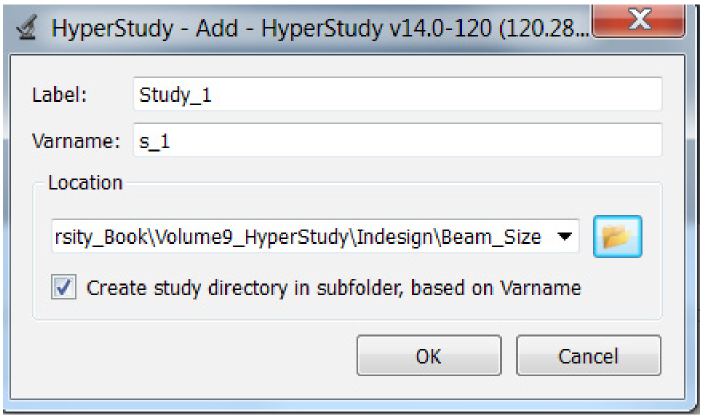
This exercise uses the model file, beam_size.fem (which should be copied to your working directory)

Model Information:

- Variables: Shell thickness
- Define responses: Mass

7.2.1 Create a New Study within HyperStudy

In HyperStudy, click on “New Study” to create a new study.



In the “HyperStudy – Add” window, click OK to use the default labels for the study label and variable.

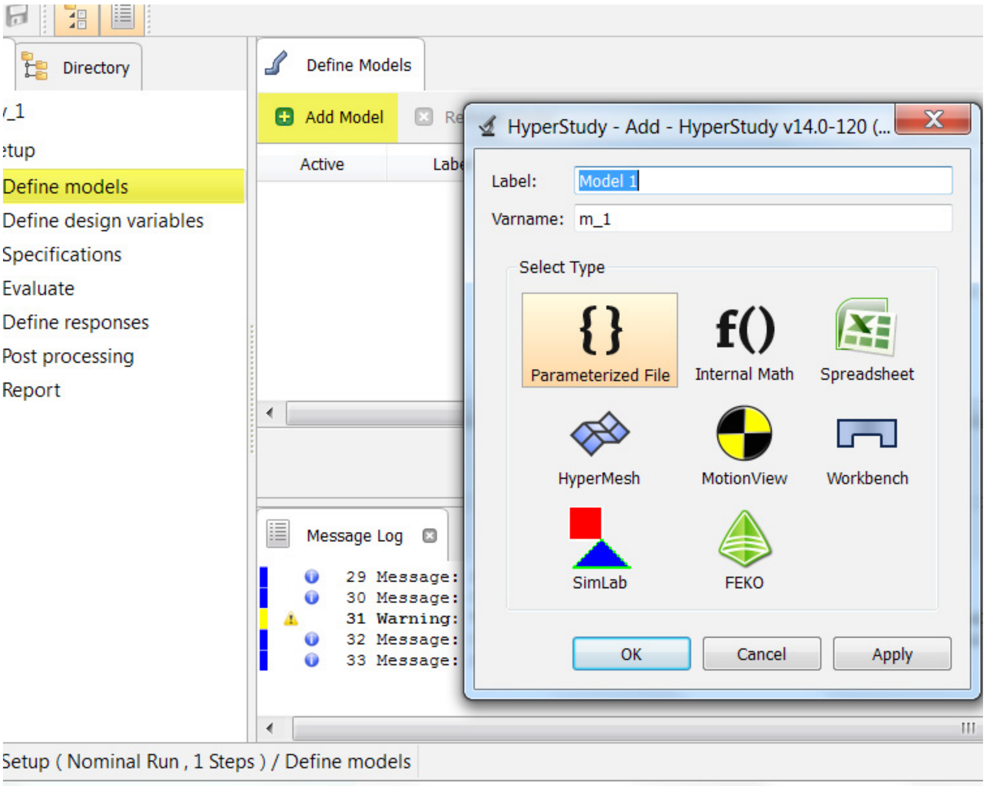
The study directory is used to contain all of your HyperStudy run files. By default, the Study Directory is your default HyperWorks user profile directory. Most users create subdirectories for their study directories according to the study or date to better organize their information.

7.2.2 Add a Model to the Study

Click “Add Model” to bring up the HyperStudy - Add dialog box.

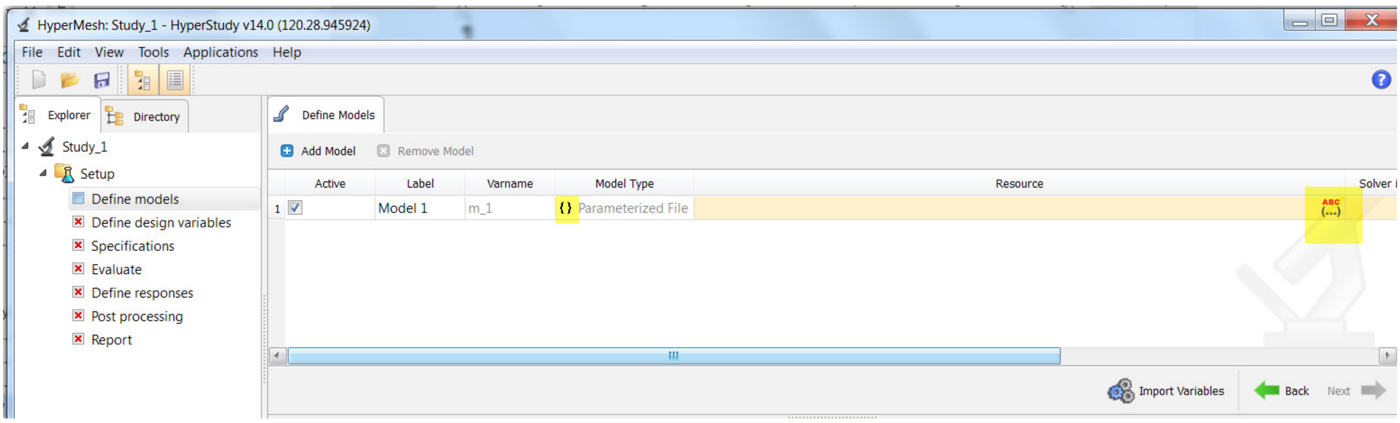
This dialog box allows you to select a model type and assign a label to this model.

Select “Parameterized File” and click OK to accept the other default values to add this new model to the study.

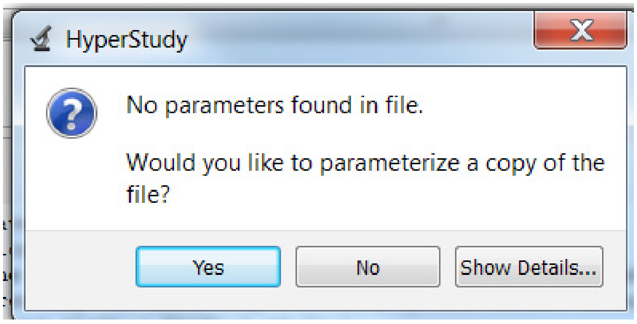


In the “Resource” tab click on “ABC …” to open the Open

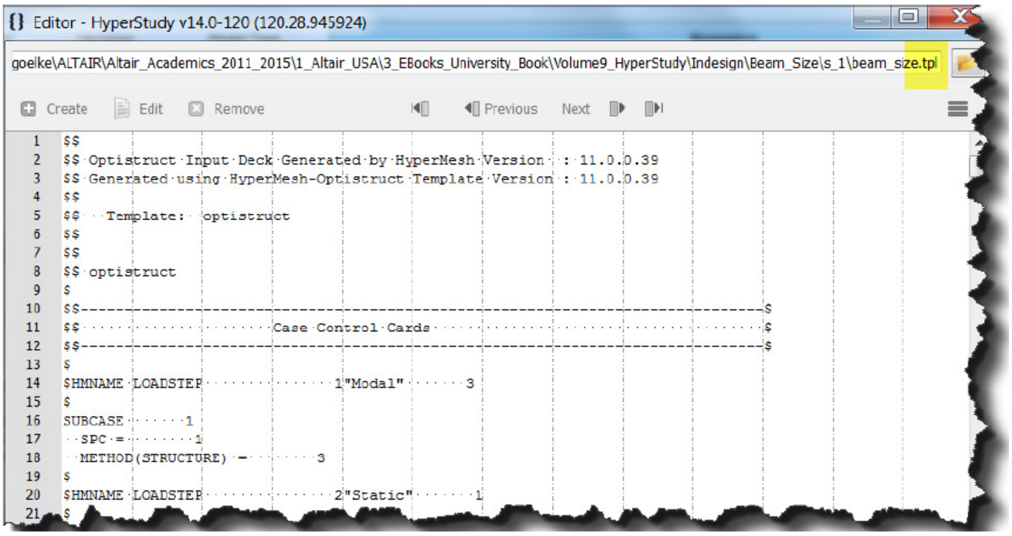
File dialog box and to load the solver deck beam_size.fem.



Note: If you load a file that does not contain parameters (as with our input file), a dialog will appear asking if you would like to parameterize the file.

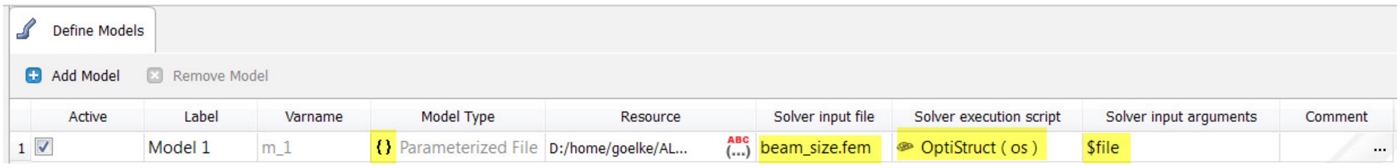


Of course, we select “Yes” to parameterize the file. The file will open in the HyperStudy Tools Editor as a *.tpl file.



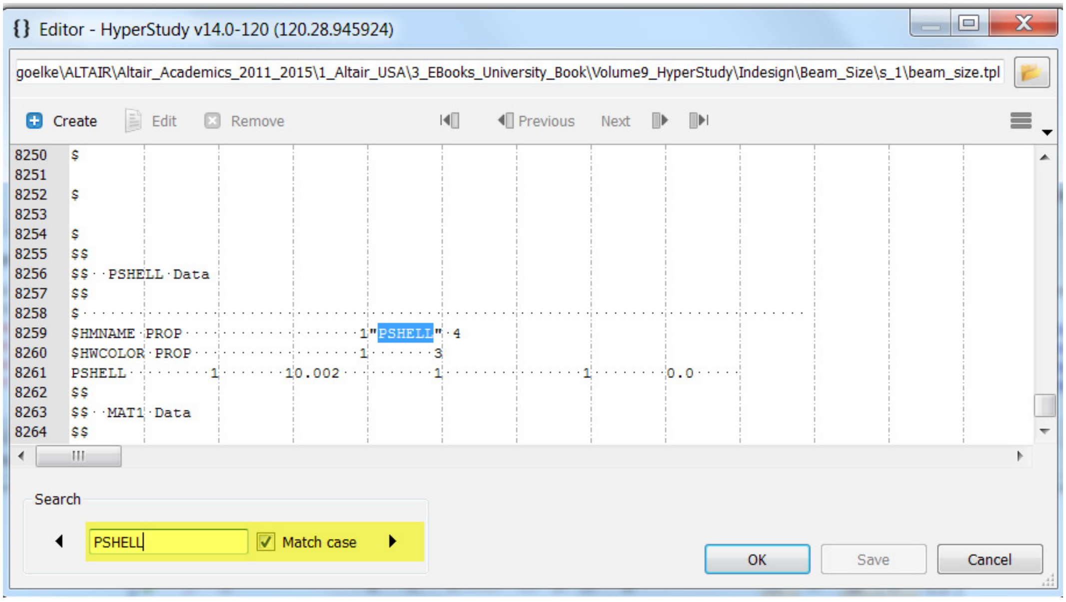
The Editor allows you to browse through the model file etc. After closing the Editor HyperStudy automatically loads the *.tpl file in the Resource column (see image below) and inserts the file name in the “Solver input file” column.

Ensure that OptiStruct (os) is input as the Solver execution script, and that the Solver input arguments is set to $file.



7.2.3 Setup the Design Variable for PSHELL Thickness

To start up the Parameter Editor dialog box click on “ABC …” in the “Resource” tab. This opens up the beam_size.tpl file. In the Search field, type PSHELL in the String to find field. Make sure to flag “Match case” checkbox and click on the arrow until you find the PSHELL Thickness card (in OptiStruct the thickness of the shell elements is specified in the PSHELL command).

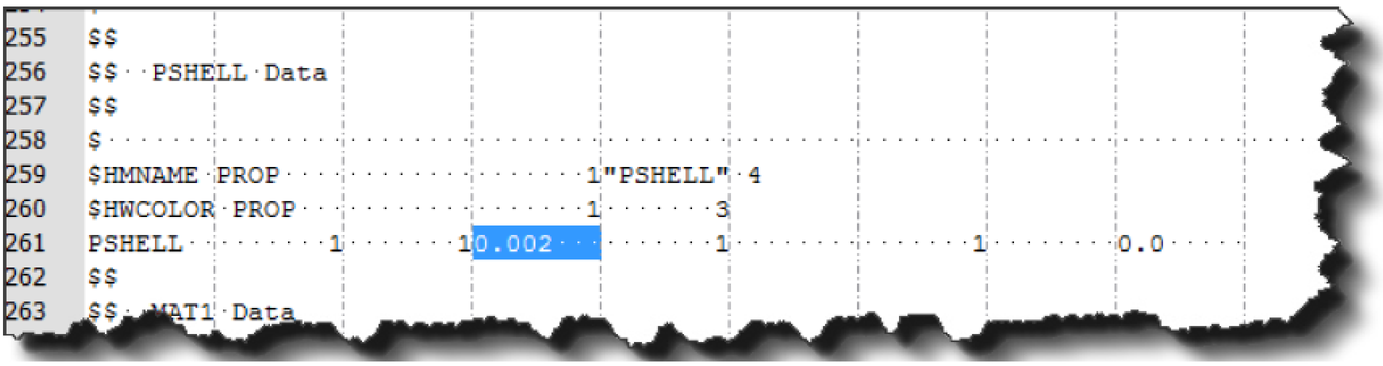


The PSHELL thickness card highlighted in the “Editor” window

The initial thickness of the shell elements is 0.002

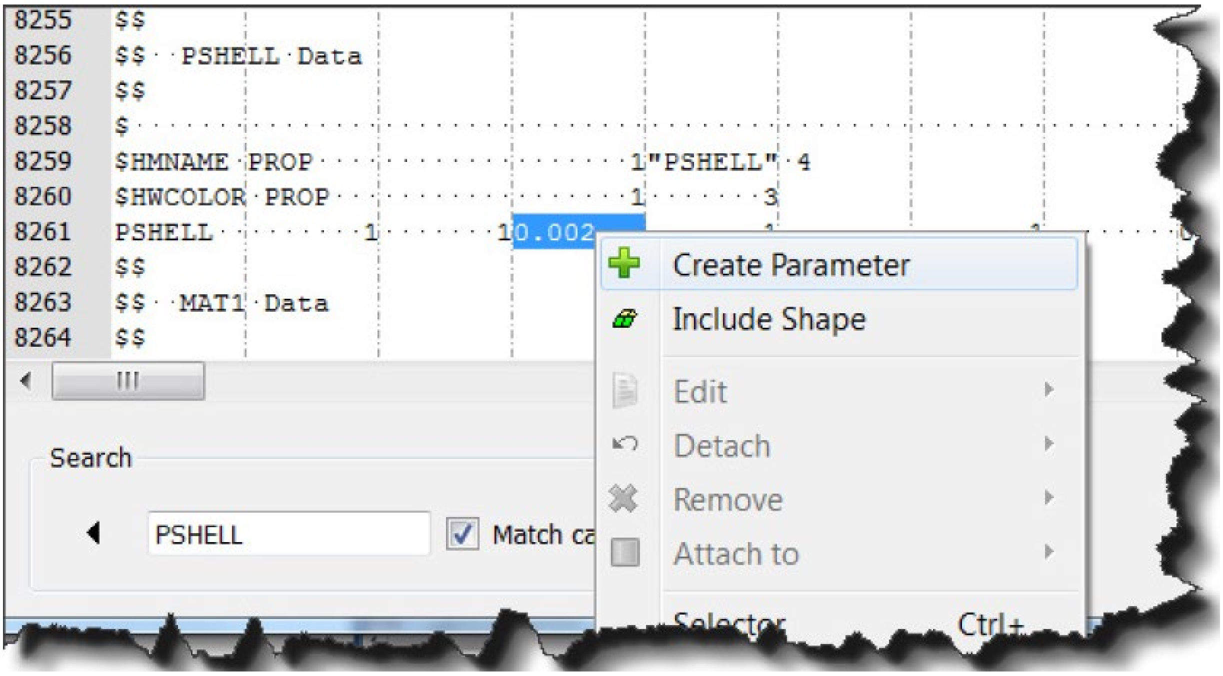
Parameterization

Within the line for the PSHELL card, select the value for the PSHELL thickness as shown below.

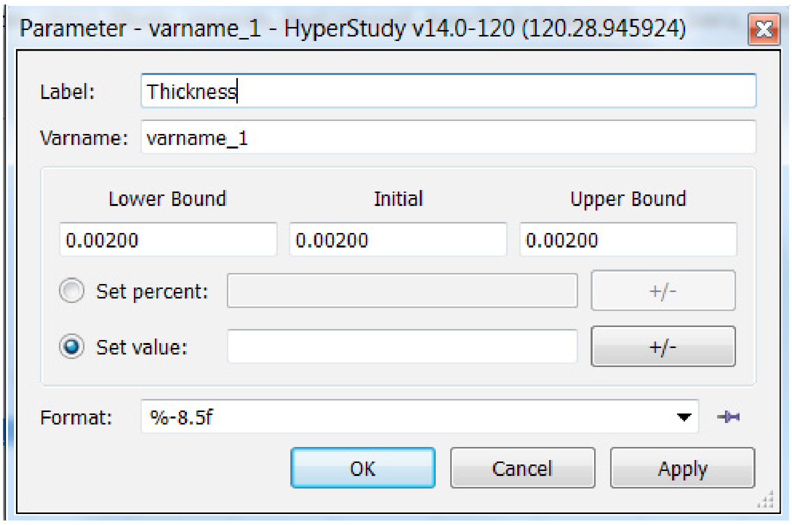


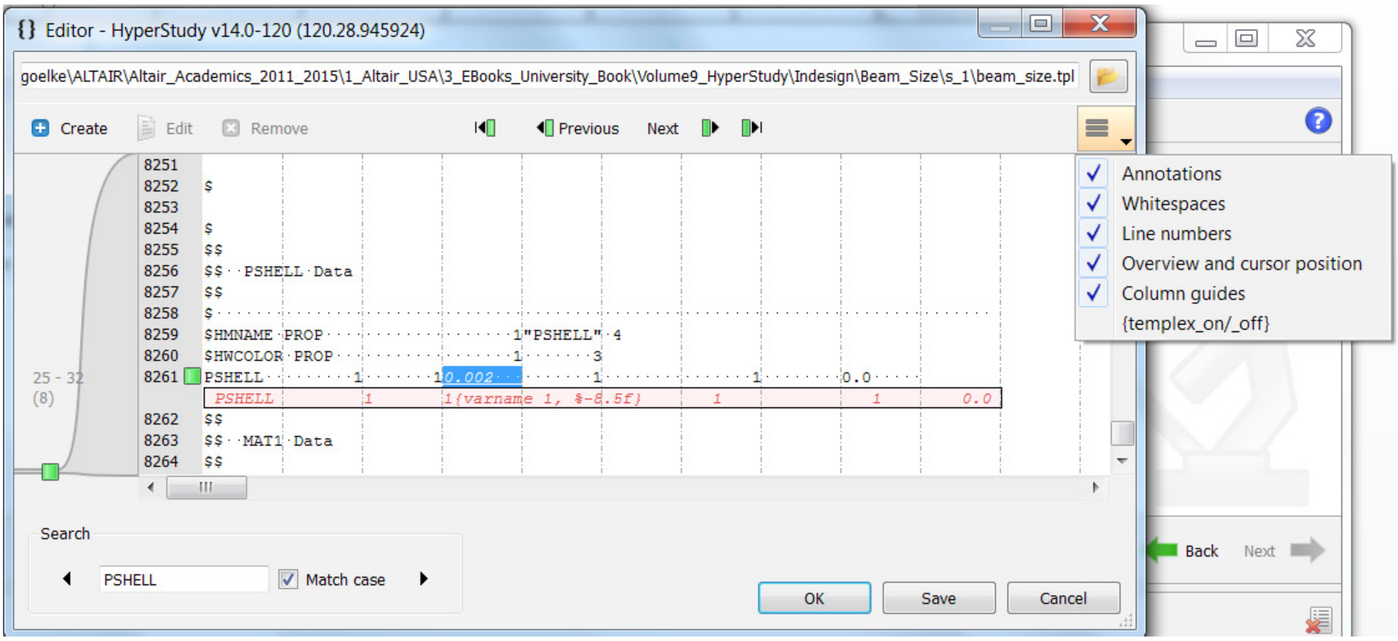
In an OptiStruct deck, each field within a card is 8 characters long. In order to select the value for the PSHELL thickness properly, select the value (0.002) as well as the three spaces following the value.

With the eight-characters selected, right-click on the selection and navigate the context menu to select "Create Parameter" to bring up the Design Variable Parameter dialog box.



In the Parameter dialog box, enter Thickness in the "Label" field. The "Format" as "%-8.5f". Then close the dialog box.

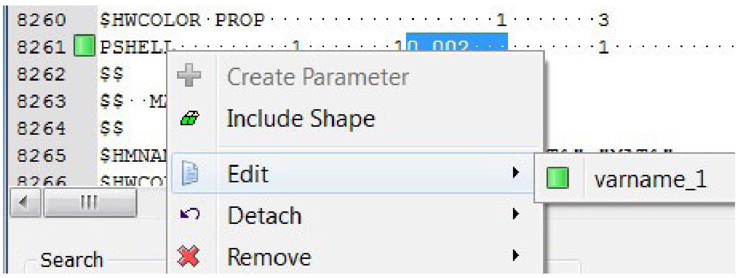




The Thickness variable now shows as Design Variable within the Parameter Editor. The entry can be edited to check/edit the values within the variable which are used within HyperStudy for design exploration.

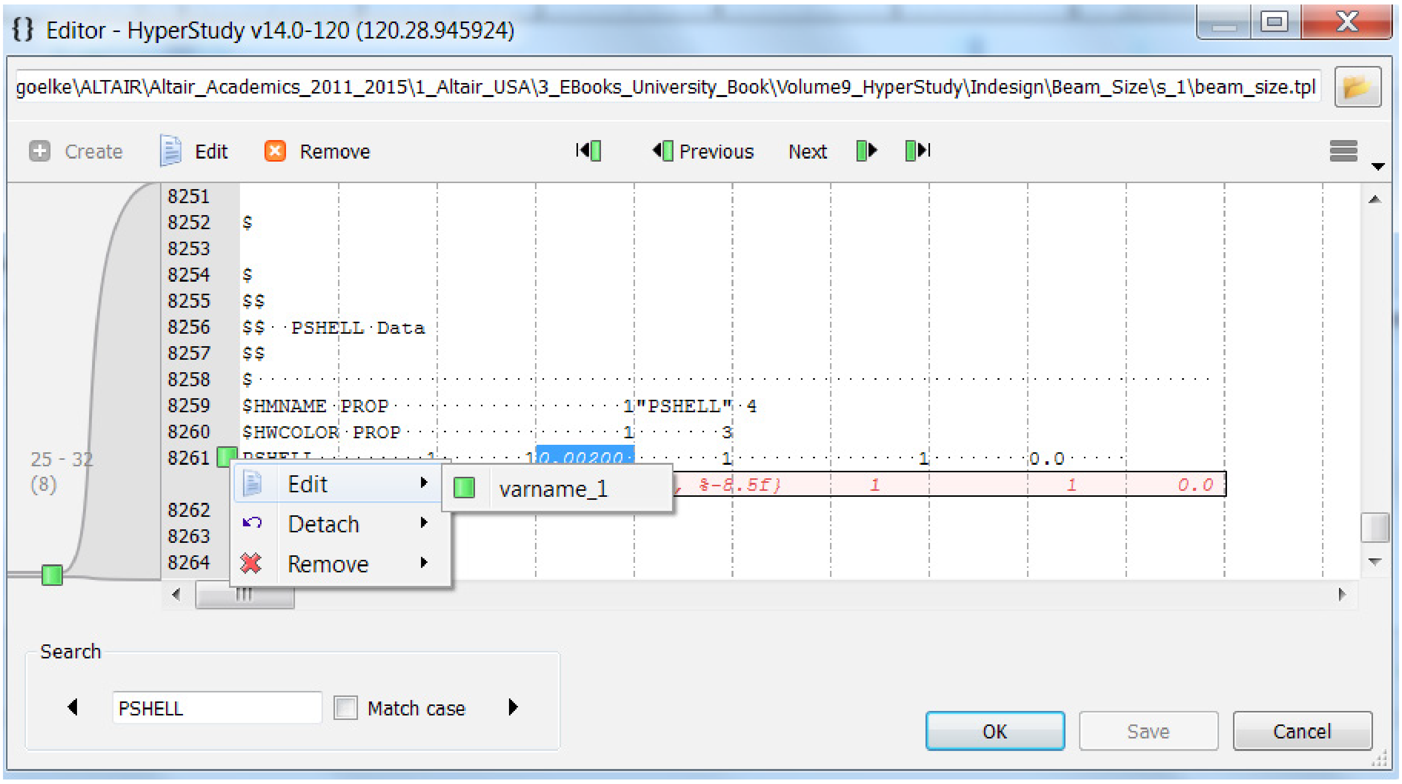
Note on “Edit”

Right click on the variable to review/edit values and select Edit from the menu list



OR

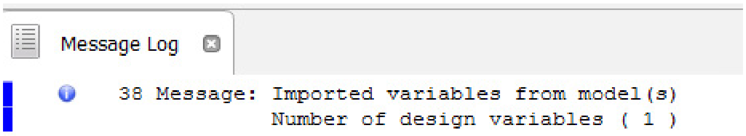
Click on the green quad and select Edit from the menu list



Click on “Save” to save this Template file as a model (beam_size.tpl). Then the window is closed by clicking on OK.

After the parametrization of the solver deck is completed (i.e. the variable is defined) the design variable needs to be imported into HyperStudy:

In the Message Log window, the information is prompted that 1 design variable was successfully imported.

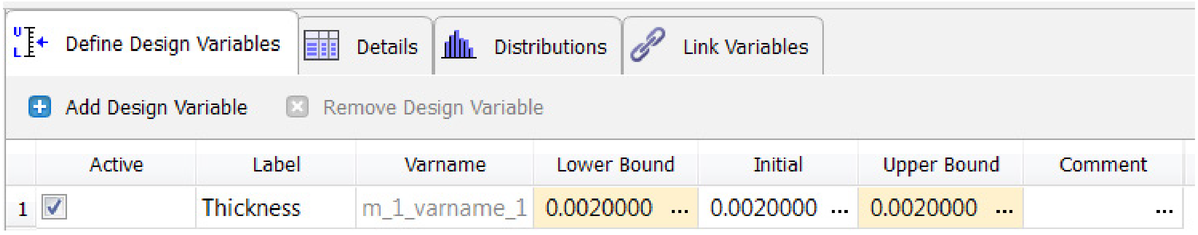


Click “Next” to proceed to the “Define design variables” (inside HyperStudy)

7.2.4 Verify Model Parameters

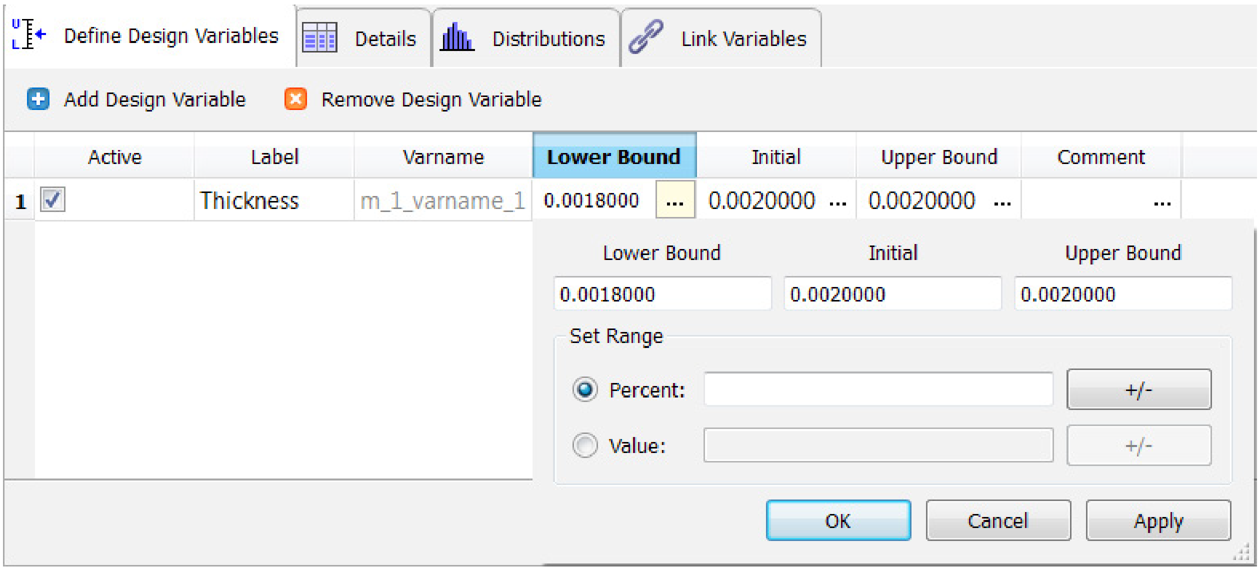
HyperStudy reads the Templex statement in the beam_size.tpl file automatically and imports the design variables.

On the HyperStudy main screen under the “Define design variables” step, HyperStudy shows the design variable that has been imported and lists it as a Template variable under Model Type.

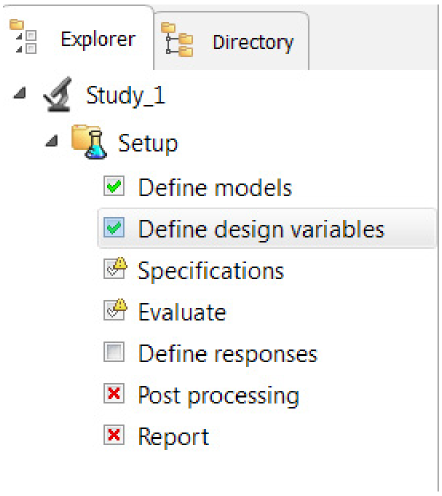


The Design Variable Linked into HyperStudy

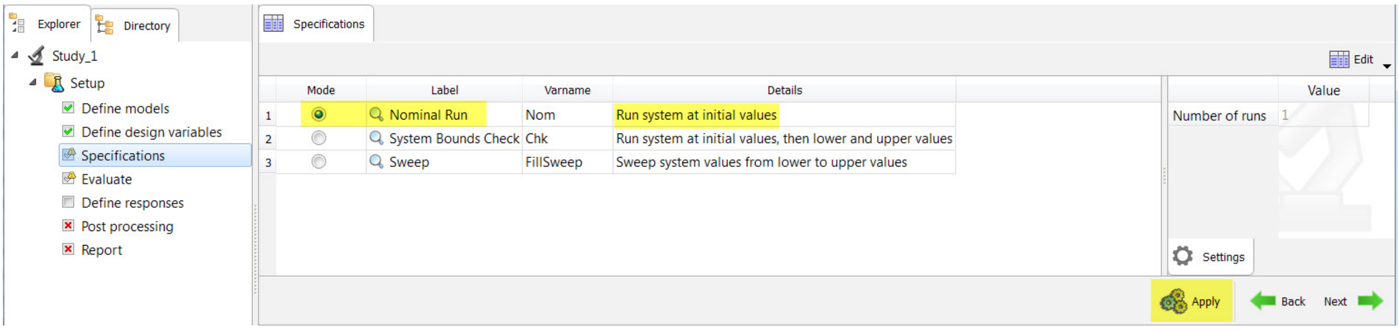
To edit the “Lower Bound” either type in the new value directly or click on “…” to open up the dialog window. As lower bound, we enter 0.0018.



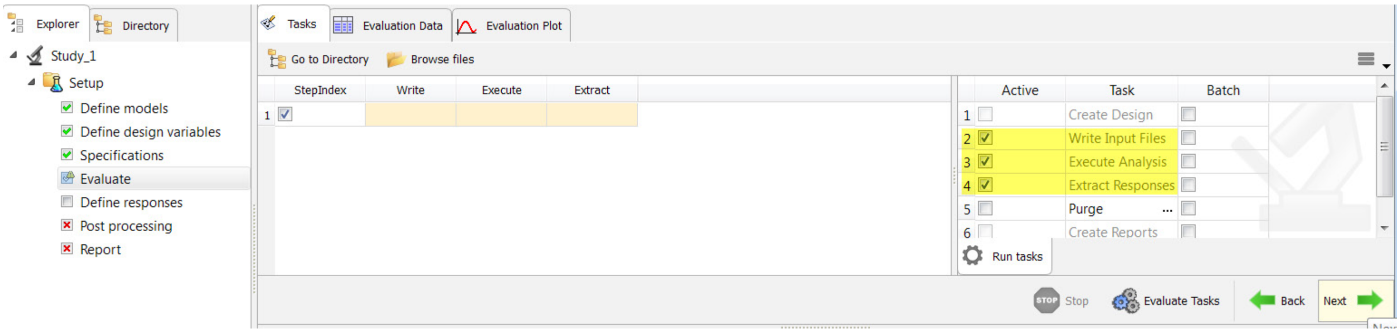
We are advancing to the “Specifications” step:



7.2.5 Submit the Nominal Run



Select the “Nominal Run” option, click on “Apply” (to accept the selection), then click on “Next”.



Make sure Write, Execute and Extract have their flag “on”, to write and execute the nominal run for this study. Eventually, to start the Nominal Run click on Evaluate Tasks.

HyperStudy writes all of the necessary input files to the study directory and executes the run using OptiStruct. The “Message Log” window at the bottom of the user interface will indicate when the model has completed execution. Once the run has completed, click Next to proceed to the Define responses step section.

7.2.6 Create the Response and Compare the Response Value

Steps are similar to the previous exercise.

7.2.7 Save the Study

In the File menu, select Save As… and assign a name to this study archive which will be saved as *hstx.

7.3 Study Setup of a Parameterized File Model with Shape Variables

The following example is kind of special:

HyperMesh will be used to create and export Shape Variables only. The model will then be parameterized in HyperStudy – similar to the previous exercise. In other words, the model will be exported from HyperMesh (HyperMesh can be closed then) and HyperStudy is started as a "standalone" product.

To better understand the differences (advantages/disadvantages) of both processes we highly recommend to study the first example as well.

Linking HyperMesh and HyperStudy, i.e. by starting up HyperStudy from within HyperMesh, as shown earlier, is extremely comfortable (and easy to use) as HyperStudy recognizes the design variables automatically. Hence no parametrizing is needed.

So, what do we gain by giving this up and parameterizing the model in HyperStudy?

In this particular case HyperMesh is dispensable.

This is especially helpful if you are dealing with rather big models (many design variables & runs) which makes it admirable to run HyperStudy in Batch-Mode on a cluster (HPC).

Hence, the focus of the example below is on how to create and export Shape design variables in HyperMesh and to parametrize the model in HyperStudy afterwards.

This exercise uses the model files, beam_shape_with_shapes_noDV.hm and beam.fem. Copy these files to your current working directory next.

Model Information:

Variables:

R1, R2, H1, H2

Define responses:

Mass

7.3.1 Creating the Shape Variables in HyperMesh

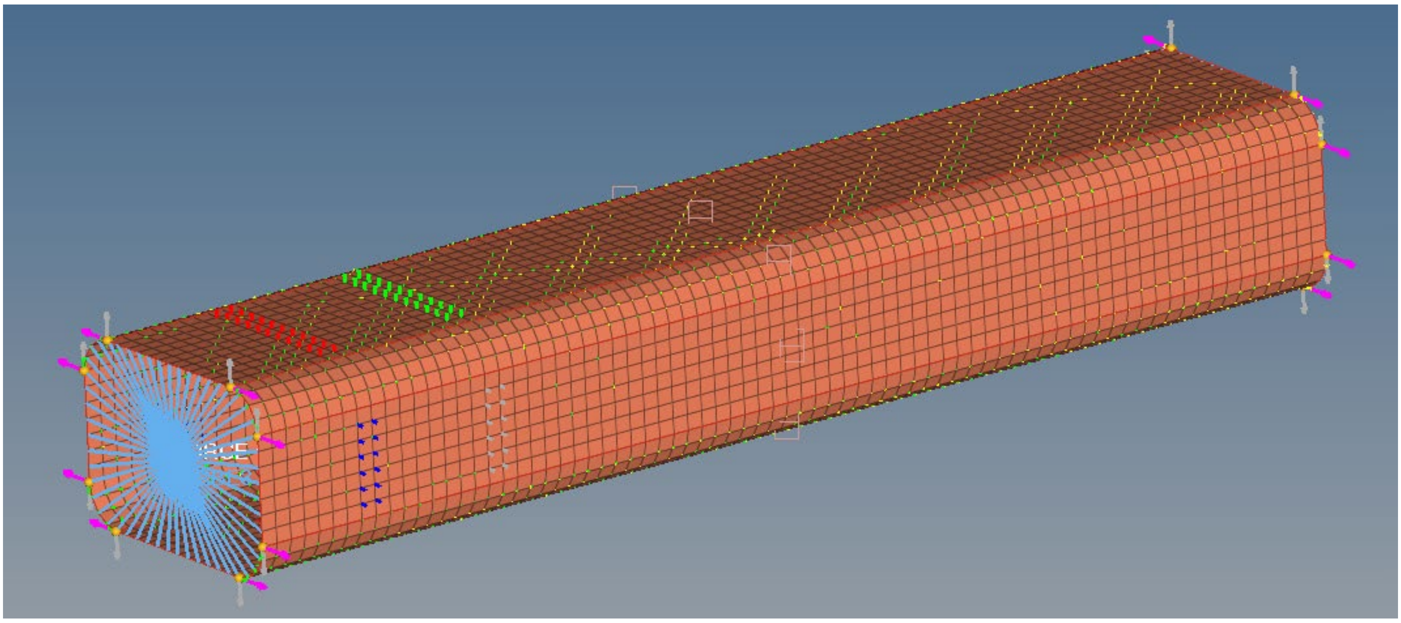
Please note, we are not going to create the shapes in the context of this tutorial. The shapes are already contained in the given model (if you like to learn more about the Morphing technology used to create the Shapes we refer you to the Online Help or/and recommend to attend a respective seminar).

The model (file) provided is an OptiStruct model. Hence, after starting HyperMesh select the User Profile OptiStruct.

Open the HyperMesh model (File→Open→Model….) named beam_shape_with_shapes_noDV.

hm

Provided you went through the first example (Study Setup for a HyperMesh Model with TH, Size and Shape Variables) you will recognize this model.

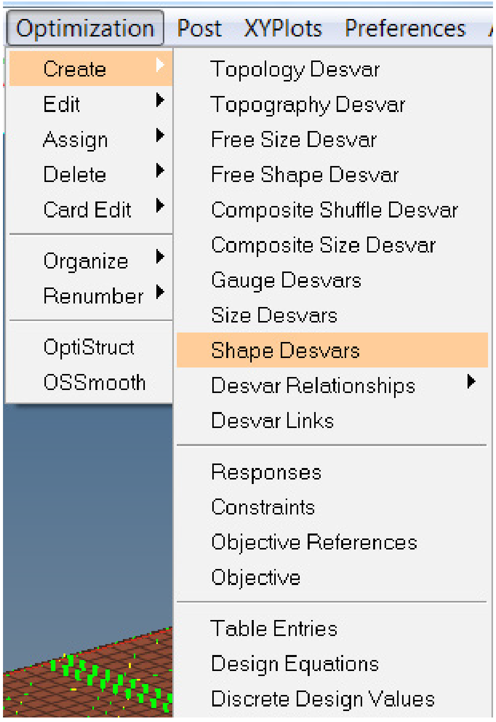


7.3.2 Creating Design Variables for the Shapes

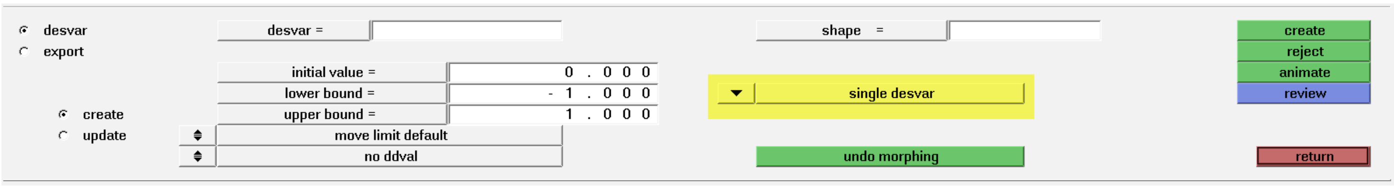
At that time the shapes contained in the model were not shape variables yet (they are part of the database but not actively used yet).

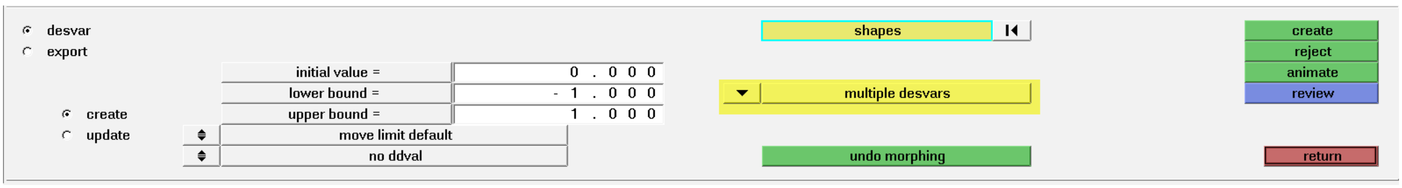
Hence, in HyperMesh Desktop, use the drop-down menu:

Optimization→Create→Shape Desvars.

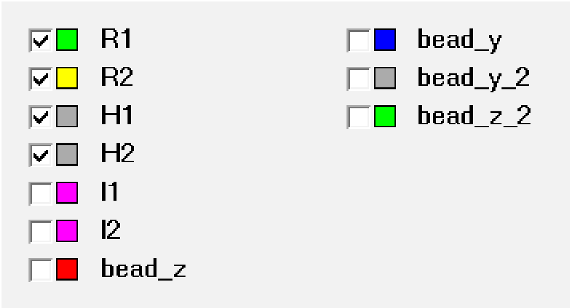


In the center of the shape panel, change the drop-down selector box from “single desvar” to “multiple desvars”.

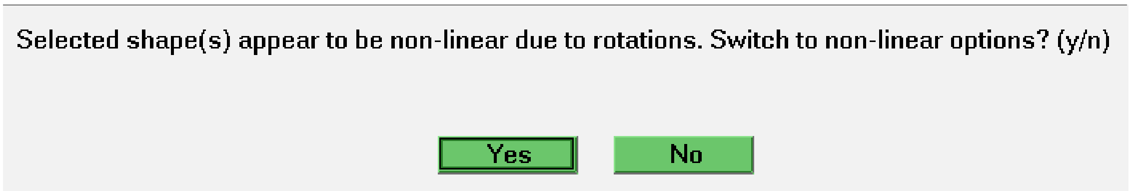




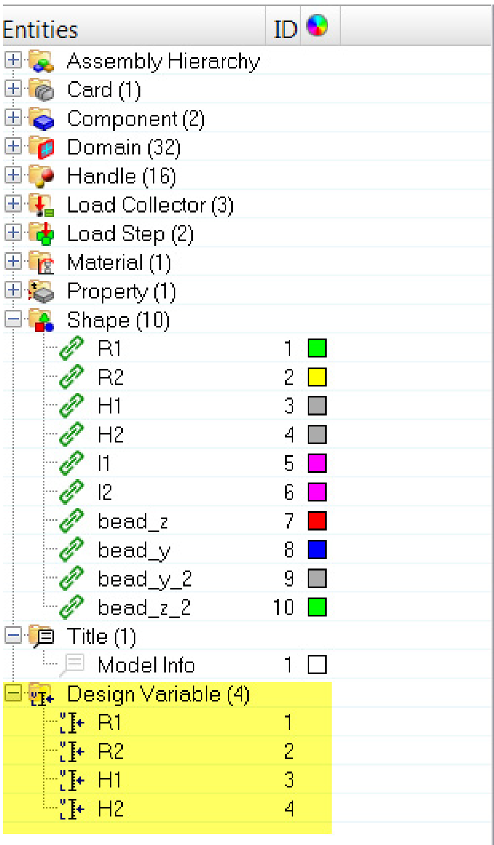
Click on the yellow “shapes” button above the “multiple desvars” drop-down selector to bring up the shape selector panel Select the shapes R1, R2, H1, and H2 as indicated in the image



Click the “create” button on the far-right side of the panel to create one design variable for each of the shapes selected. HyperMesh Desktop will bring up a dialog box asking if you wish to change to non-linear shape design variable options because of rotations in the shapes. Click “No” to create these shape parameters as linear variables.



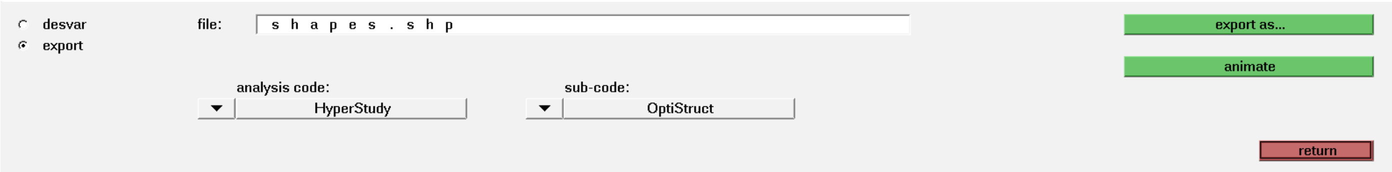
The newly created design variables R1, R2, H1, and H2 are listed in the Model Browser.



7.3.3 Export the Design Variables as a .shp File

On the left-hand side of the shape panel, switch the selector from desvar to export. This allows us to export the grid perturbations for the shape displacements in a number of formats.

- Switch the “analysis code”: selector to HyperStudy and “sub-code”: to OptiStruct



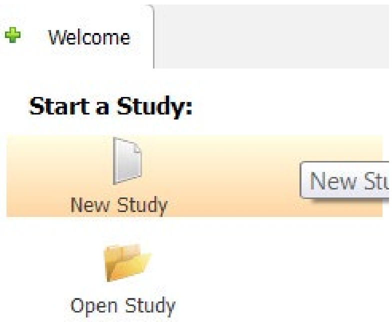
- Edit the filename in the “file” field to be “shapes.shp”
- Click “export as…” to save the file

This completes the working steps in HyperMesh (HM can be closed). The actual study set-up is in HyperStudy.

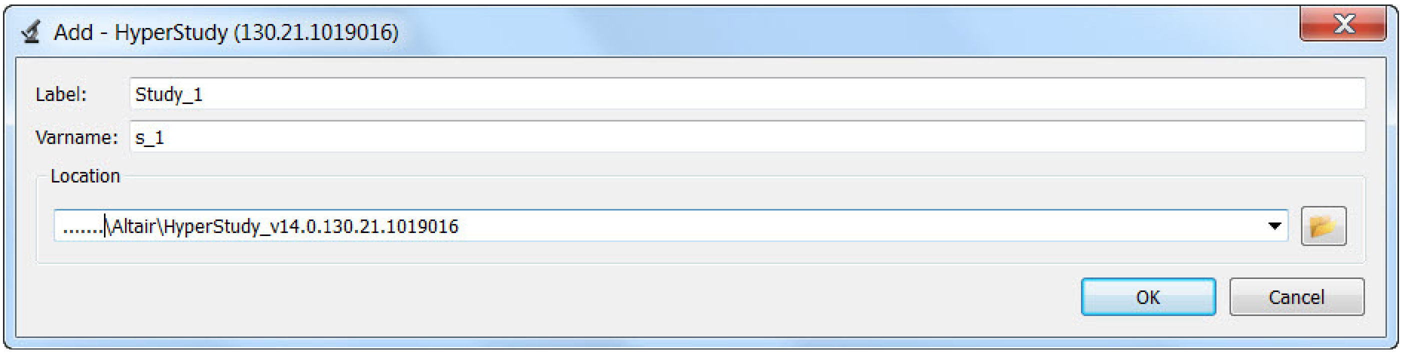
7.3.4 Create a New Study within HyperStudy

This time HyperStudy is started from within the Start Directory (Windows).

In HyperStudy, click on “New Study” to create a new study



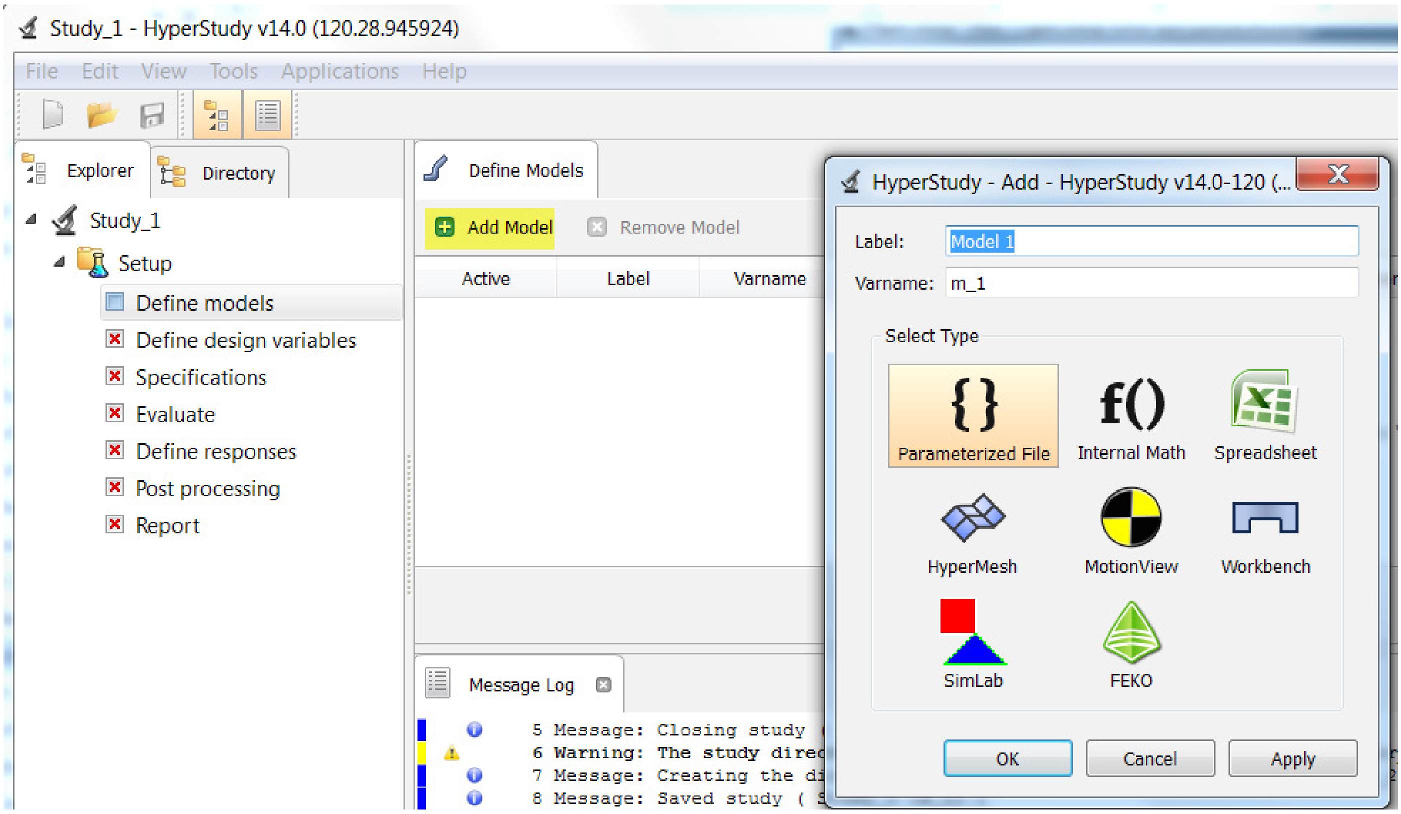
In the HyperStudy - Add window, click OK to use the default labels for the study label and variable.



Proceed to the “Define models” step

7.3.5 Add a Model to the Study

Click “Add Model” to bring up the “HyperStudy – Add” dialog box

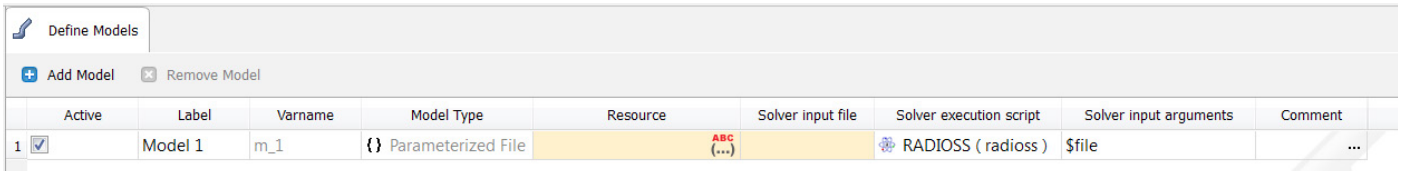


As before - This dialog box allows you to assign a name label to this model and change the variable type as well as specifying the application origin of the model.

As we are parameterizing the model in HyperStudy, “Select Type” is set to “Parameterized File”, accept the other default values.

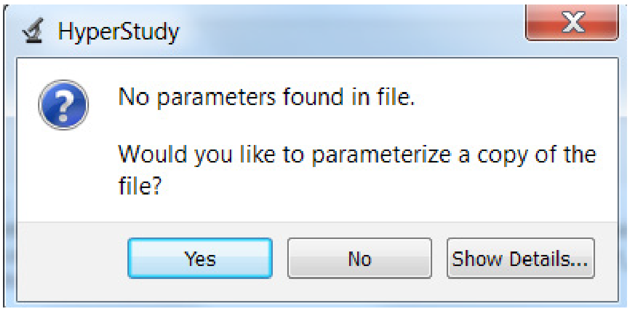
Of course, we need to import the regular FE model into HyperStudy. In this regular file (ASCII) the shape variables exported from HyperMesh will be embedded.

Import of the regular FE Model:

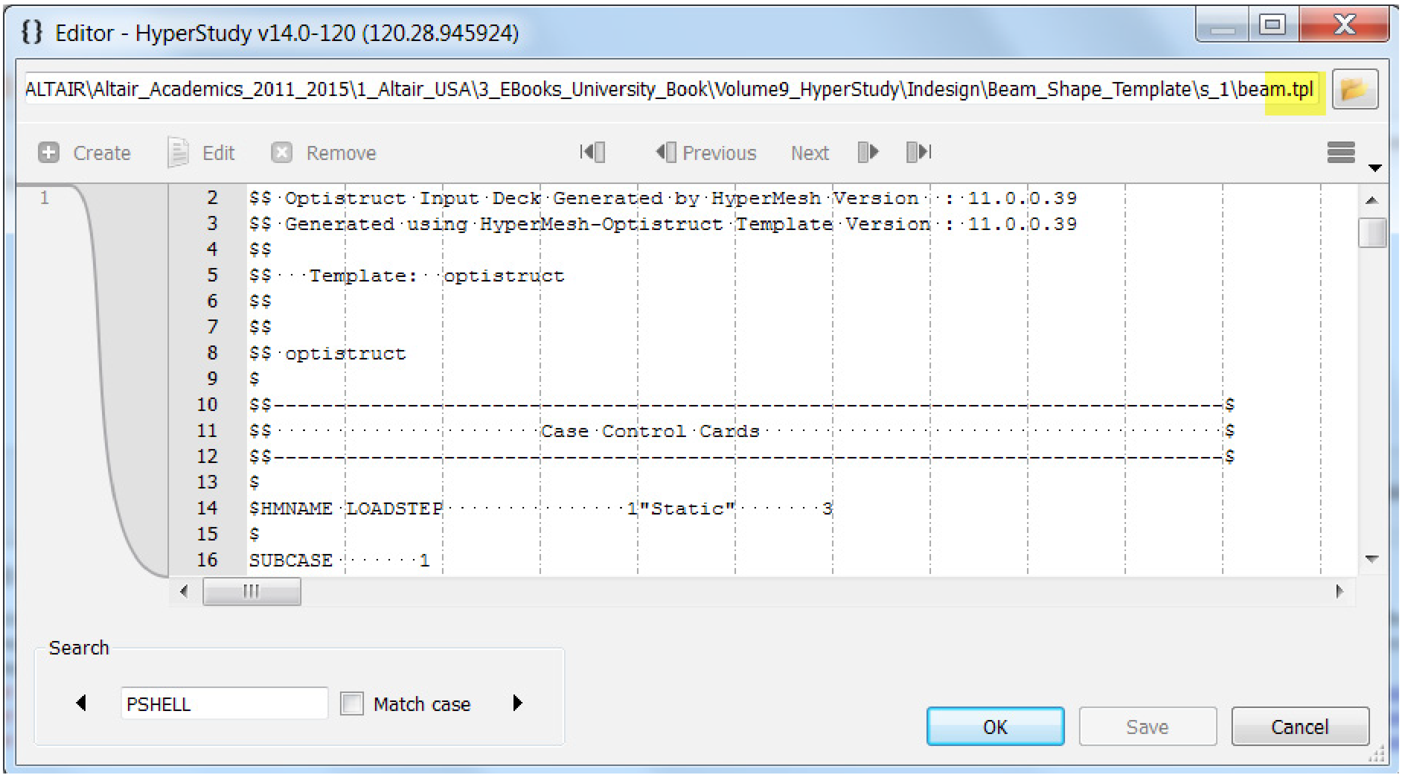


In the “Resource” tab open the Open File dialog box (ABC …) and navigate to file beam.fem

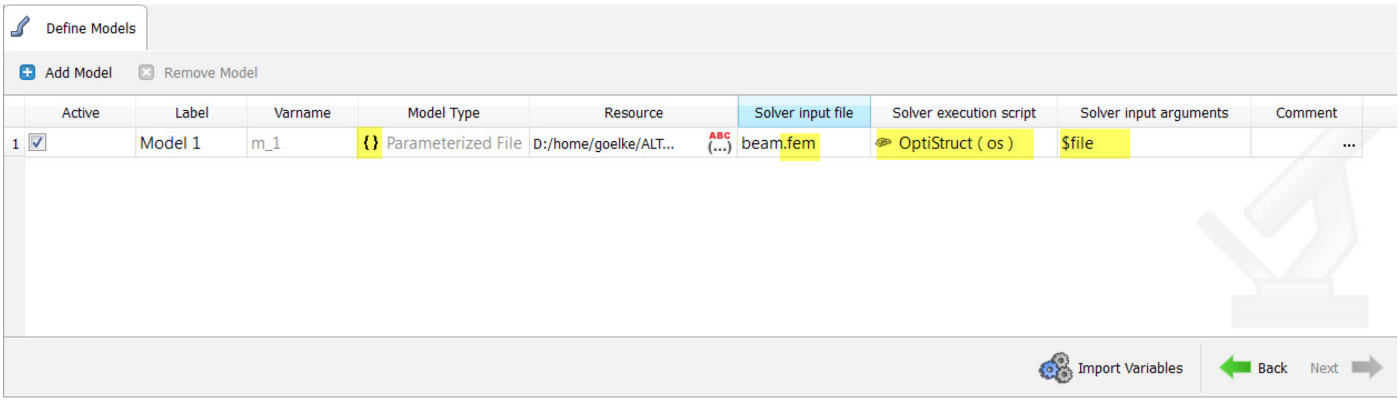
If the referenced /loaded file does not contain parameters, a dialog will appear asking if you would like to parameterize the file.



Select “Yes” to parameterize the file, the file will open in the HyperStudy Tools > Editor… as a .tpl file (here beam.tpl).



Ensure that OptiStruct (os) is input as the Solver execution script, and that the Solver input arguments is set to $file.

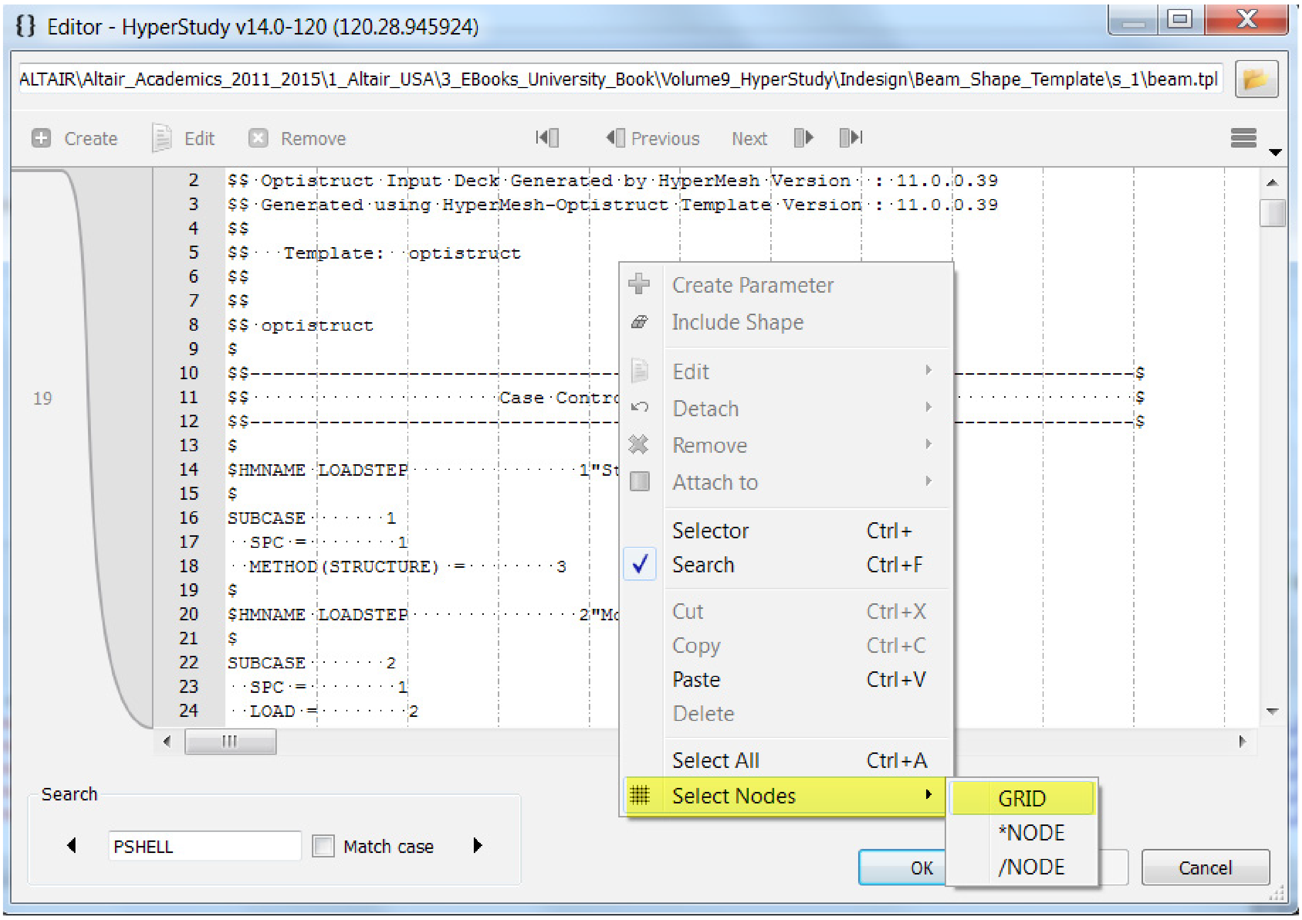


Now it's time to add the shape variables to the model FEA deck (beam.tpl)

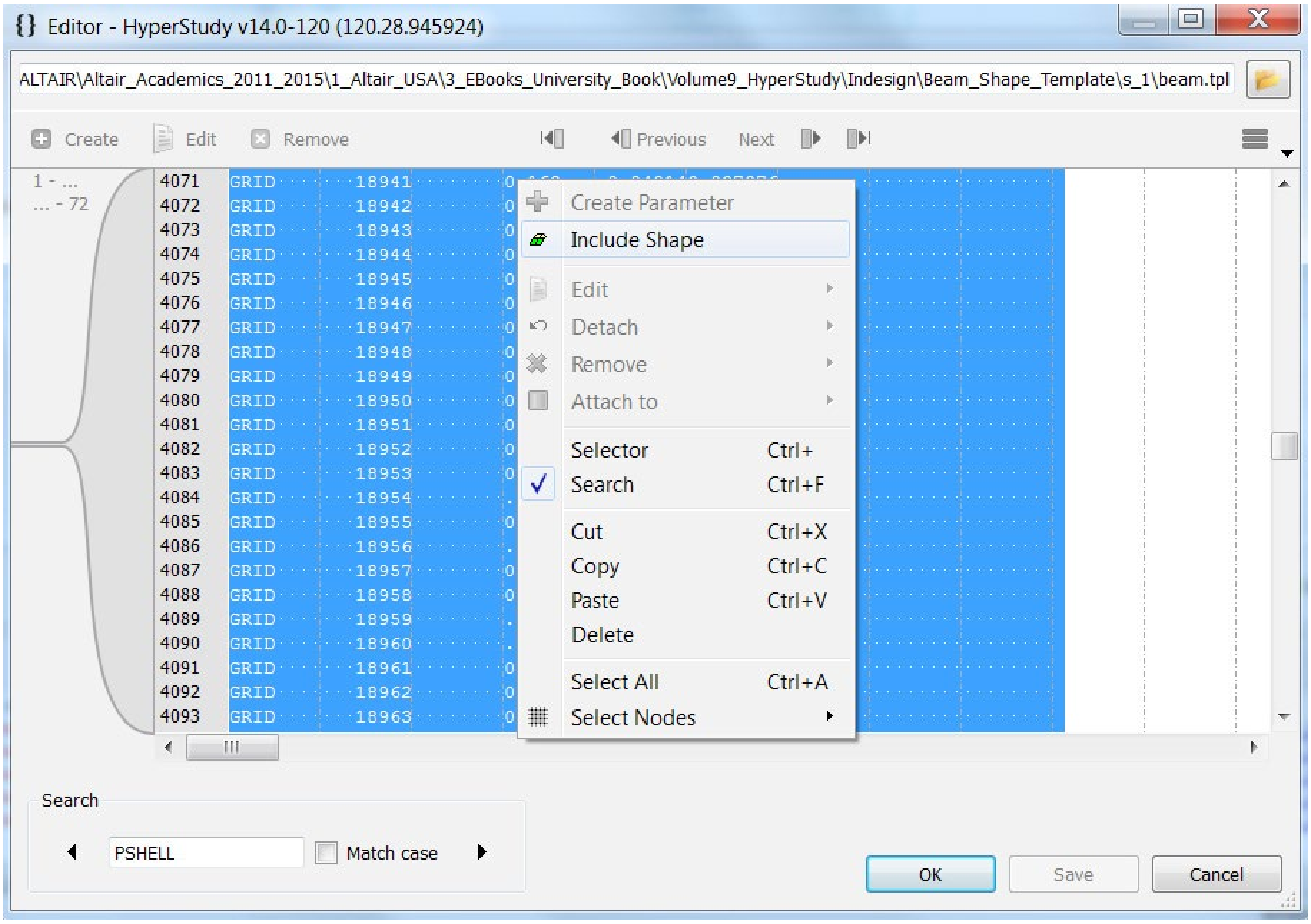
In the Resource tab click on "ABC" again, which then opens up the Parameter Editor.

7.3.6 Set up the GRID Points to Create a Shape Template

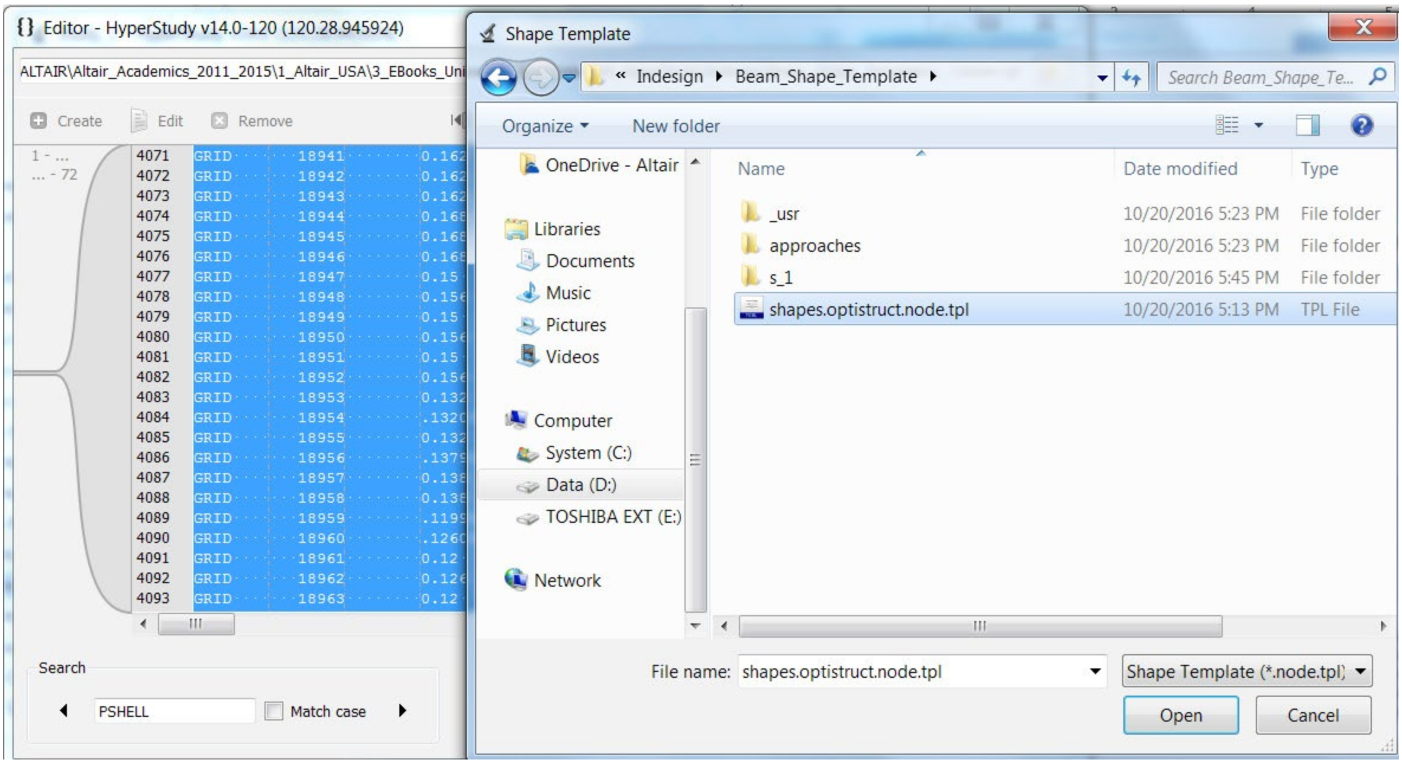
In the Parameter Editor dialog box (right mouse click), "Select Nodes" → GRID. This automatically selects all lines starting with the keyword GRID in the beam.tpl file (here: line 45 to line 4150).



With the GRID cards selected, right-click on the selected area and navigate the context menu to select "Include Shape"

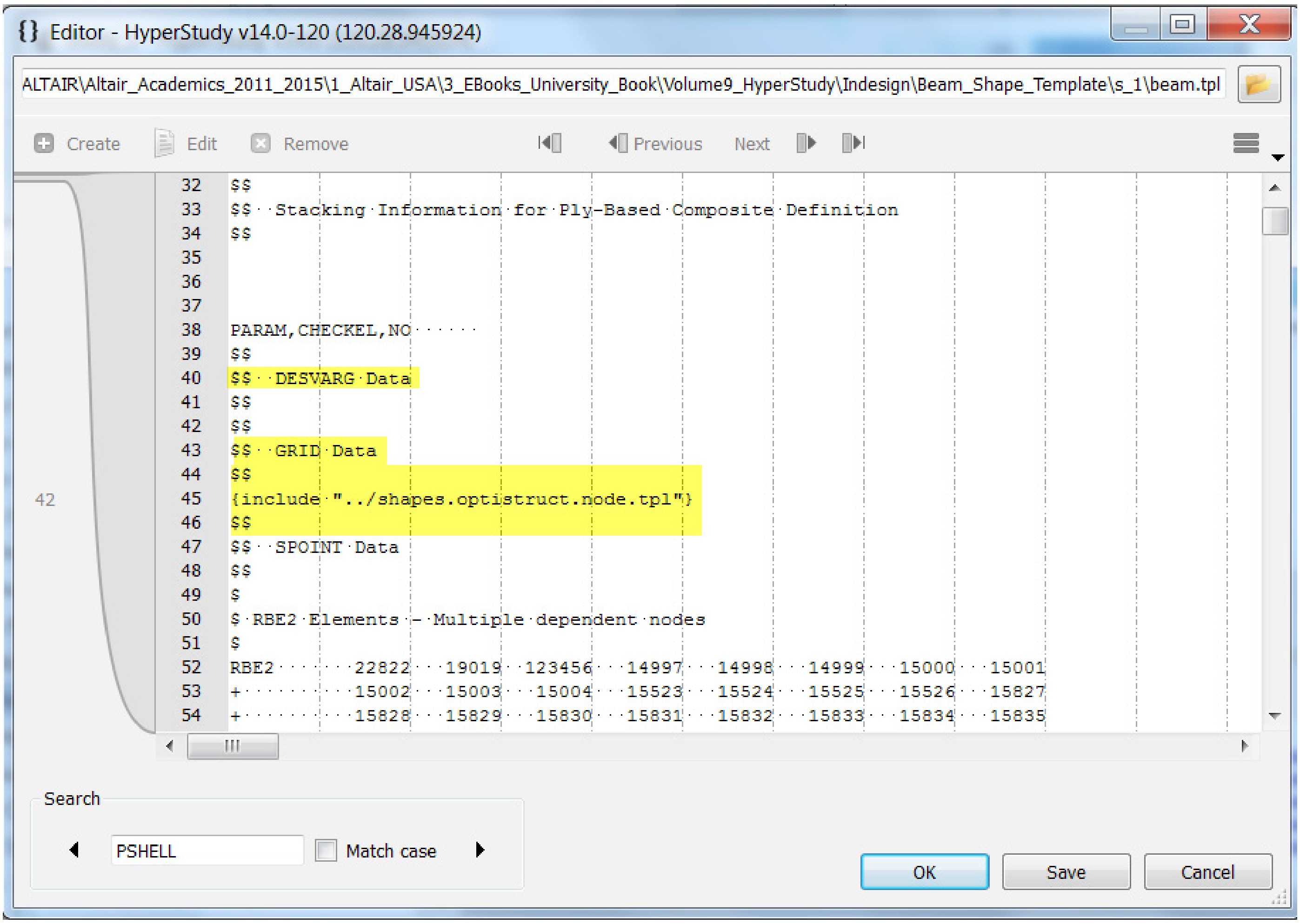


In the "Shape Template" dialog box, navigate to the location where the "shapes.shp" file was saved. Open the "shapes.optistruct.node.tpl" file in that directory.

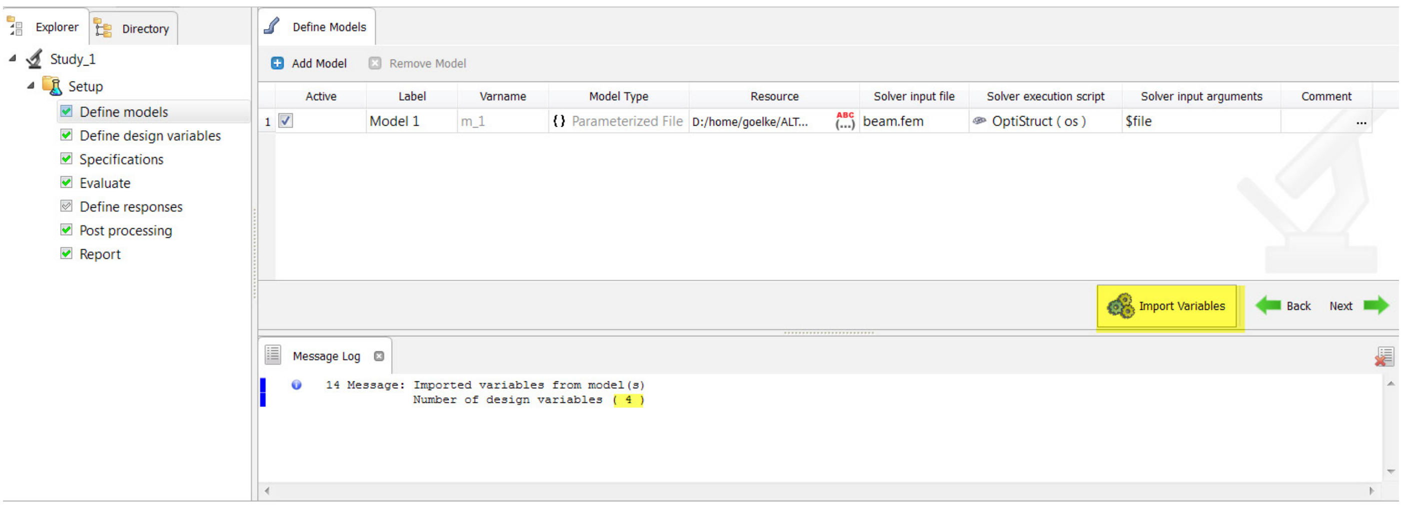


This replaces the nodes in the model with the parameterized shapes, and the shapes are imported into the list of design variables.

In other words, all the GRID data are replaced by the include file which we referenced in the previous step.



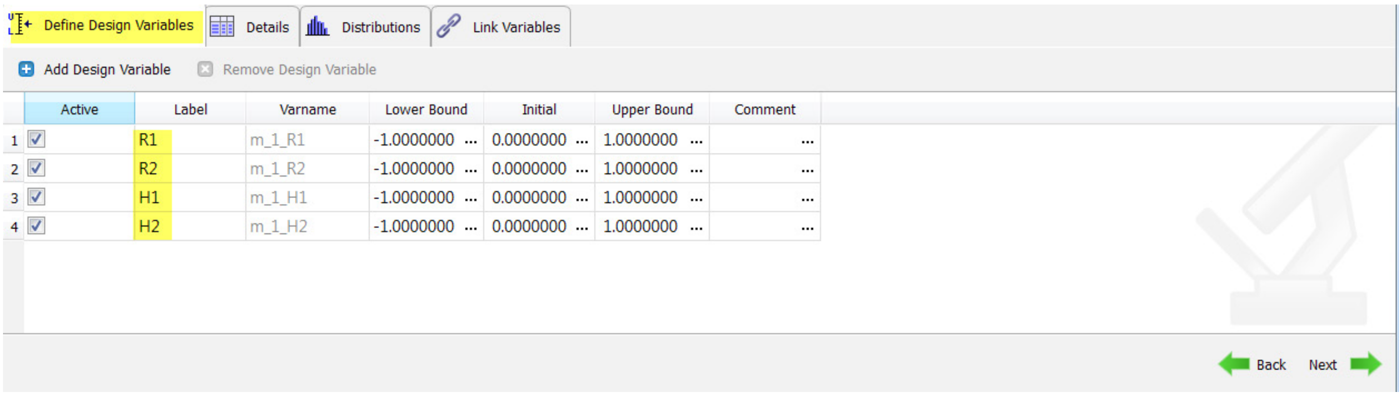
Click on Save to save this Template model (beam.tpl), then select OK to exit and return to HyperStudy. Next step is kind of exciting again. Will the Design Variables be "recognized" and imported in HyperStudy? To import the variables to HyperStudy, click on "Import Variables" (at the bottom of the Work Area).



Looks good. Great. We can proceed with the next step.

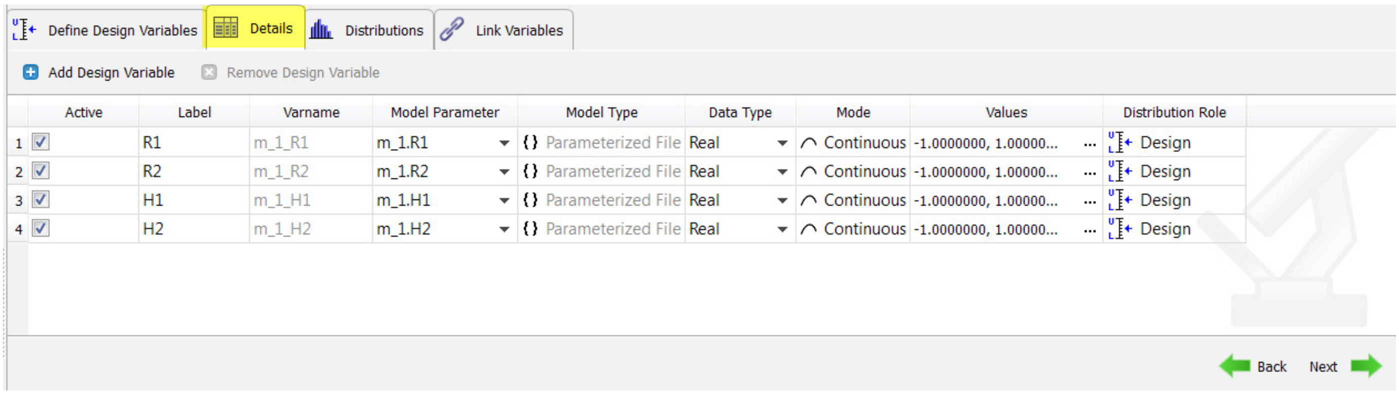
7.3.7 Verify Model Parameters

HyperStudy reads the Templex statement in the beam.tpl file automatically and imports the design variables.



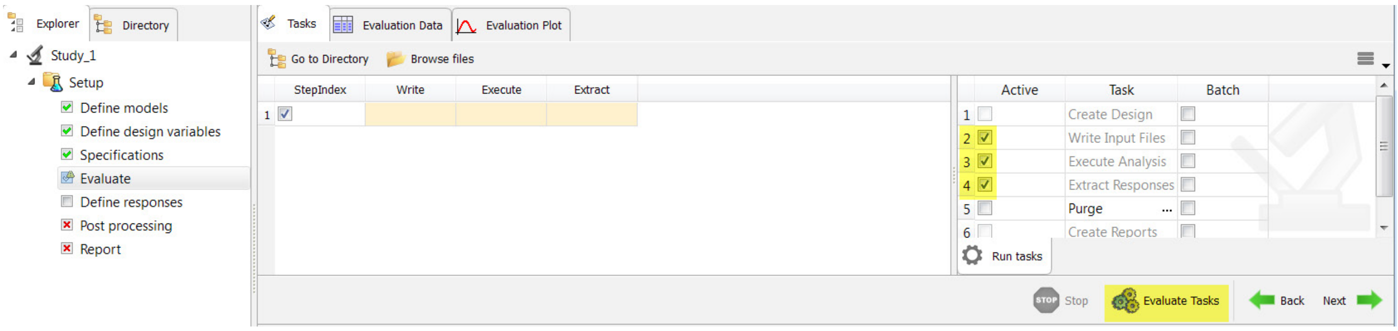
Under “Define Design Variables”, HyperStudy shows the design variables that have been imported.

More information about the design variables are listed under “Details”.



And eventually, we are ready to start the Nominal Run

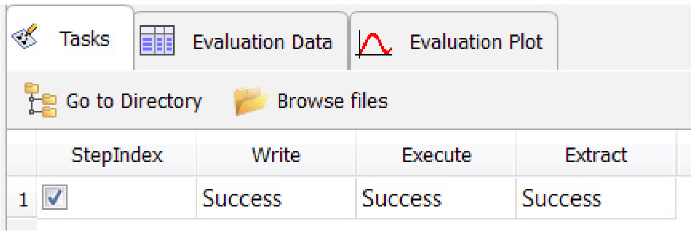
Click Next to advance to the “Specifications” step to run the Nominal Run.



7.3.8 Submit the Nominal Run

Make sure Write, Execute and Extract have their flag “on”, to write and execute the nominal run for this study. Click on Evaluate Tasks and then Next.

HyperStudy writes all of the necessary input files to the study directory and executes the run using OptiStruct. The Messages window at the bottom of the user interface will indicate when the model has completed execution.



Once the run has completed, click “Next” to proceed to the “Define responses” section.

7.3.9 Create the Response and Compare the Response Value

To complement the study setup, we need to define responses (same process as in the other examples; included in here for your convenience).

We are going to look at the response of type Mass. Information about the Mass is contained in the beam.out file.

Define Responses:

- The steps are same as the previous exercise.

7.3.10 Save the Study

In the File menu, select Save to save the current Study. If this study has not yet been saved, a Save As… dialog box will pop up requesting a name for the Study archive.

Save the study archive in the working directory using the file format *.hstx

7.4 DOE Method Comparison: Arm Model

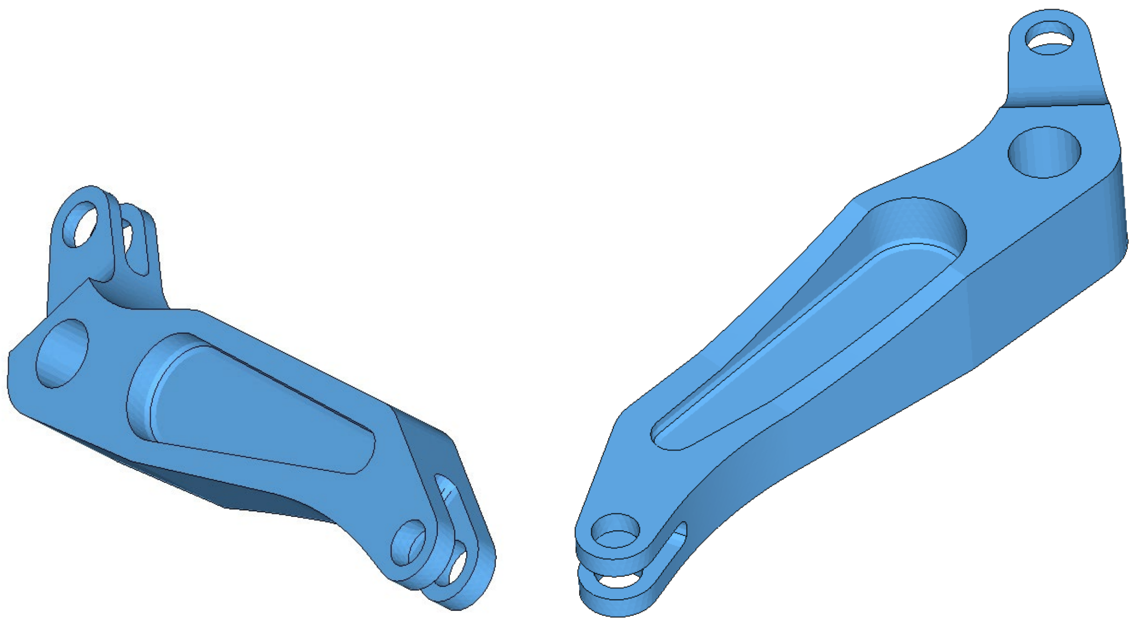
In this quite comprehensive example, the following aspects will be addressed:

- Set up a Full Factorial DOE
- Set up a 2-Level Resolution III, IV and V Fractional Factorial DOE
- Set up a 3-Level Fractional Factorial DOE
- Set up a Plackett-Burman DOE

Compare these methods in accuracy and efficiency

The files used in this example are

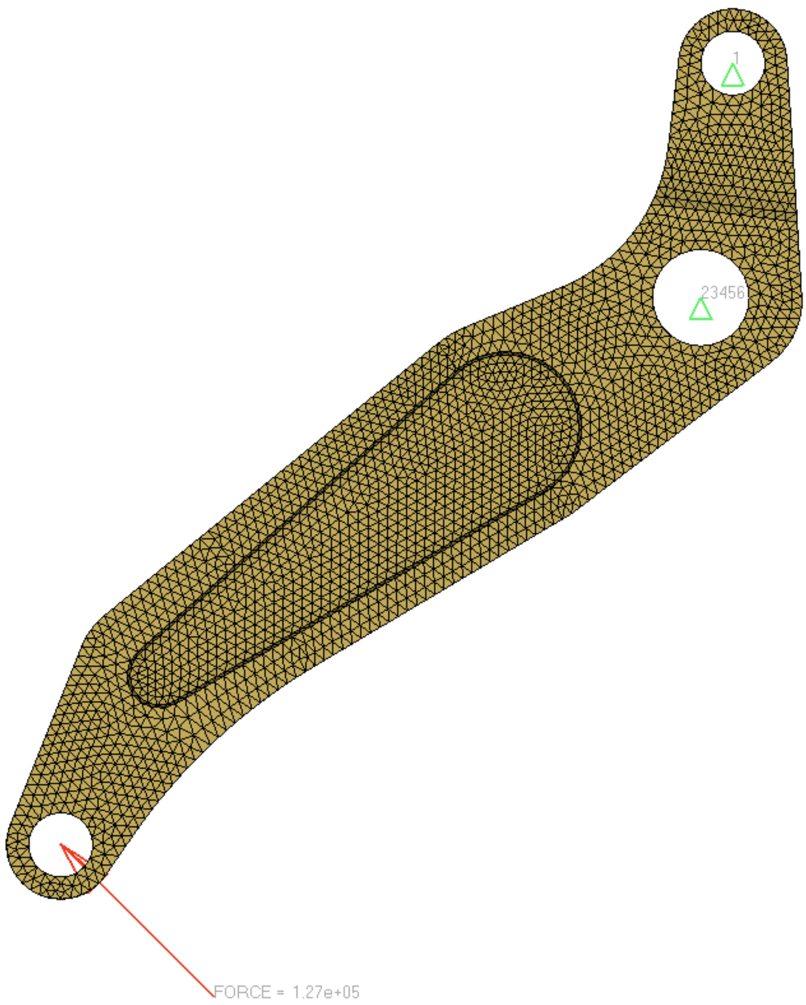
- crank_morph.hm
- arm_model.tpl Template file
- arm_model.optistruct.node.tpl Grid coordinates template
- arm_model.shp Grid perturbation vector data for arm_model.optistruct.node.tpl



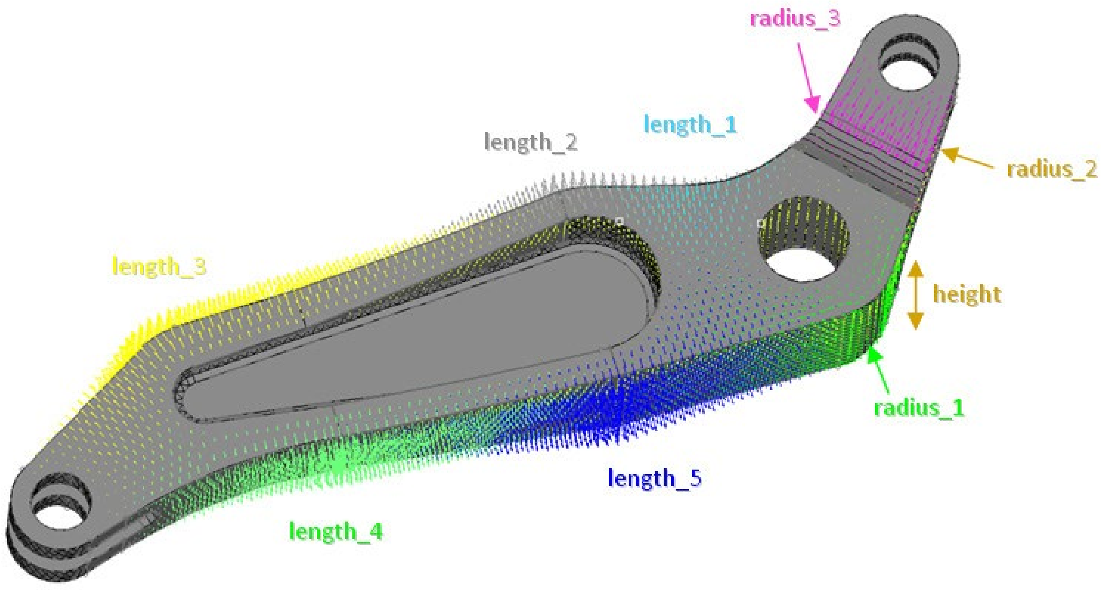
7.4.1 Problem Definition

The arm shown below is clamped at one end and under an axial loading on the other end. This model has been meshed and modeled in HyperMesh. Linear static analysis is performed using HyperWorks finite element solver, OptiStruct.

The current (nominal) design gives a volume of 1.7667E06 mm3 , a maximum displacement of 1.41 mm and a maximum stress of 195.29 MPa. The designer can change the following properties of the arm: 6 shape variables for the overall length and height of the part, 3 shape variables for the three radii (see image below).



Concentrated load and boundary conditions



Shapes defined on the “arm model”

The design can change shape in nine different regions, as shown on the image above, and five responses are of interest as detailed below.

Shape design variables:

Length1	**Lower Bound = -0.5, Initial Bound = 0.0, Upper Bound = 2.0**
Length2	Lower Bound = 0.0, Initial Bound = 0.0, Upper Bound = 2.0
Length3	Lower Bound = -1.0, Initial Bound = 0.0, Upper Bound = 1.0
Length4	Lower Bound = -1.0, Initial Bound = 0.0, Upper Bound = 1.0
Length5	Lower Bound = -1.0, Initial Bound = 0.0, Upper Bound = 1.0
Radius1	Lower Bound = -2.0, Initial Bound = 0.0, Upper Bound = 2.0
Radius2	Lower Bound = -0.5, Initial Bound = 0.0, Upper Bound = 1.0
Radius3	Lower Bound = -0.5, Initial Bound = 0.0, Upper Bound = 1.0
Height	Lower Bound = -1.0, Initial Bound = 0.0, Upper Bound = 1.0 Three

Three responses:

- Volume (mm3)
- Max Von Mises Stress (MPa) over all the model
- Max. displacement of all nodes of the model (mm)

7.4.2 Problem Setup

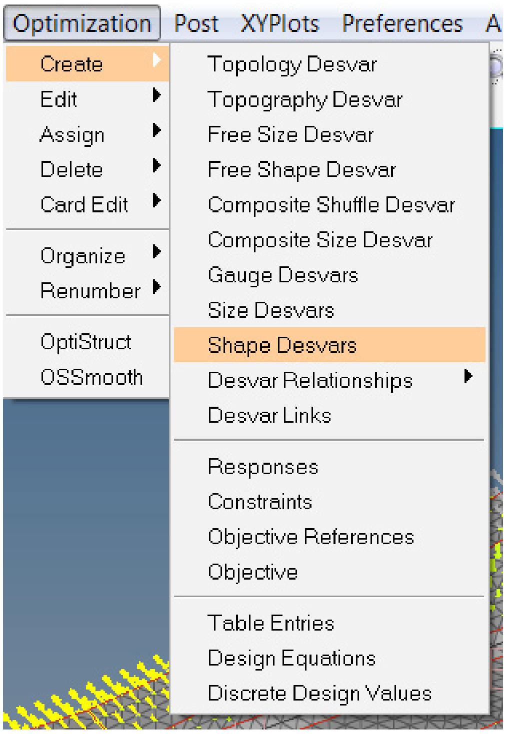
The nine different shapes (not shape design variables) are created with HyperMorph, a tool for doing parametric mesh-based shape changes (morphing), contained in HyperMesh (In the morphing process, domains are created around the FE model features. Handles are defined on the edges of these domains. By moving a handle, nodes in the associated domain will also move. Each of these movements can then be saved as shape variables and can directly be exported to HyperStudy)

In this exercise we don't show how morphing works but rather refer to the many Online Help tutorials about Morphing. We start, like in the other example above, by creating (exporting) shape design variables from within HyperMesh.

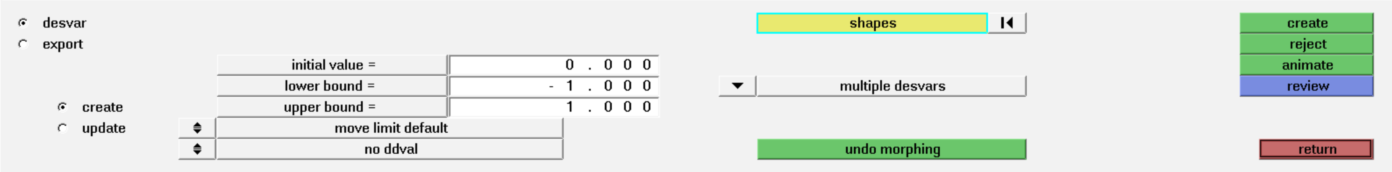
Export the Shape Parameterization from HyperMesh

Review the crank_morph.hm model file in HyperMesh (make sure the OptiStruct User Profile is turned on).

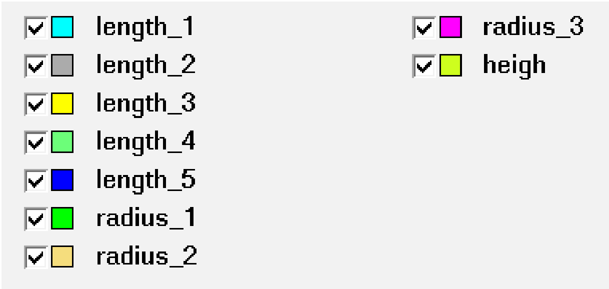
To export the Shape Design Variables, go to Optimization→Create→Shape Desvars



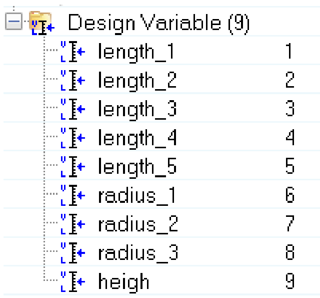
Make sure that "multiple desvars" are selected.



The lower bound and upper bound are set to -1 and 1 and initial value to 0. (i.e. all the shape desvars will have the same bounds). Select all shapes (clicking on the yellow “shapes” panel)

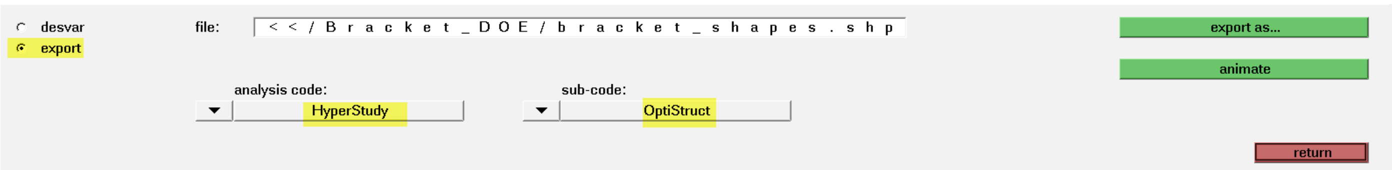


and conclude this step with “create”. This creates a shape design variable for each shape selected. If you want to animate the shapes, you can click the animate button.



The shape design variables will now be used in HyperStudy. However, we first need to export the variables.

Activate the radio button “export”, set the “analysis code” to HyperStudy (as we are going to use the variables in HyperStudy), and “sub-code” to OptiStruct (the solver we are using in this study).



The file extension must be *.shp (here the file name is bracket_shapes.shp).

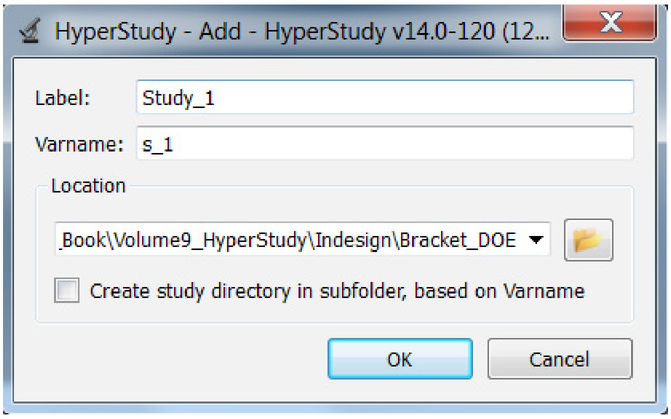
The following two files are written:

- bracket_shapes.optistruct.node.tpl (Grid/node coordinates template)

- bracket_shapes.shp (Grid perturbation vector data for bracket_shapes.optistruct.node.tpl)

(Depending on your preferences, HyperStudy can be started from within HyperMesh. In that case all design variables would be readily available in HyperStudy. Alternatively, HyperStudy can be started separately and the design variables are then “built” in the study).

In the context of this example HyperStudy will be started as a standalone. The working directory is named Bracket_DOE.



7.4.3 Define Models - Parametrization

The Study setup follows the same standard procedure as described in some detail in the other examples:

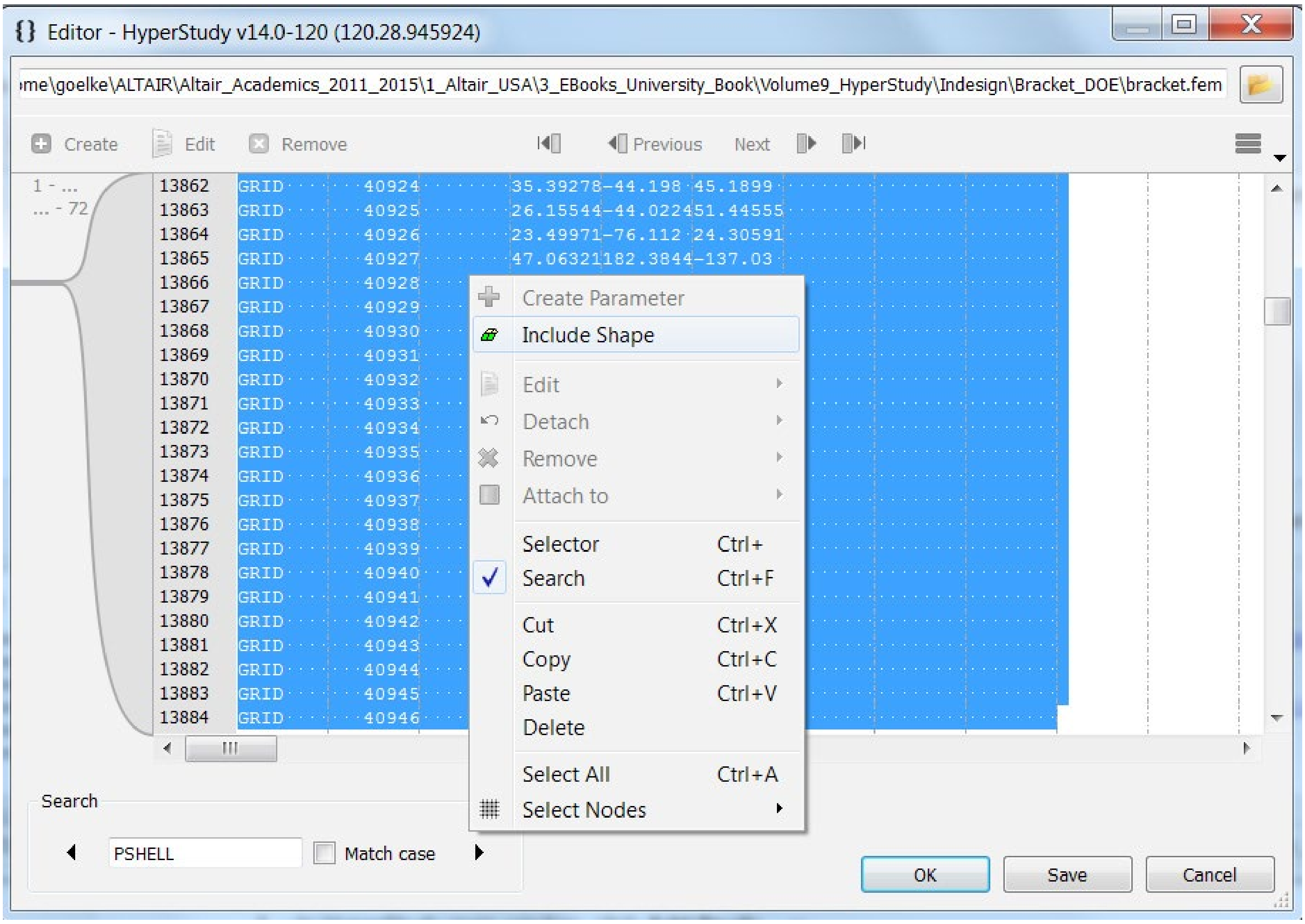
Define Models

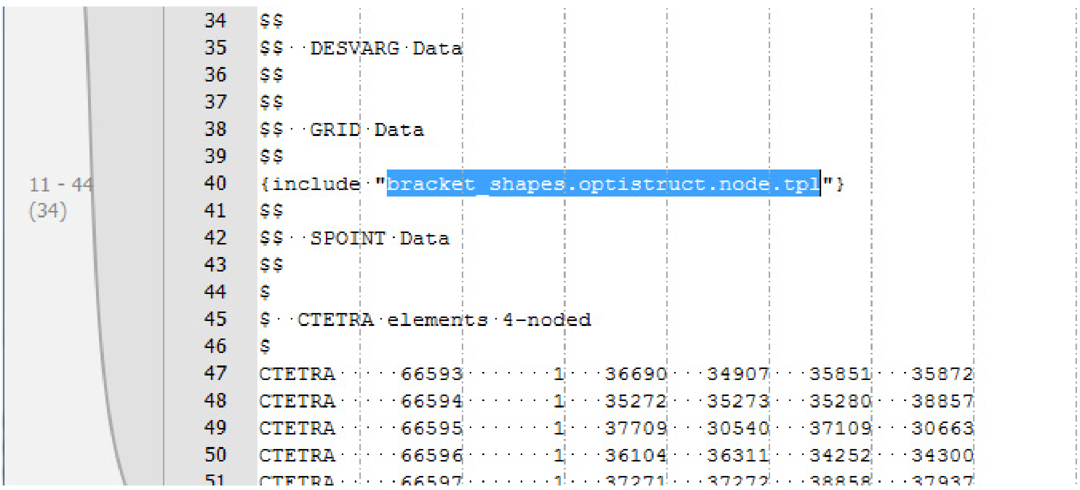
- Type: Parameterized file
- Resource File: bracket.fem
- Solver input file: bracket_DOE.fem (doesn’t exist yet)
- Solver execution script: OptiStruct (os)
- Solver arguments: $file

Specifying the resource file (bracket.fem) opens up the Editor.

All Grids (nodes) will be replaced by the shape design variables exported from HyperMesh before. Utilizing “Select Nodes →GRIDS” will select all GRIDS in the ASCII file.

Right mouse click opens the dialog window shown in the image below: Select “Include Shape” to replace all GRIDS by an Include file (here: bracket_shapes.optistruct.node.tpl)

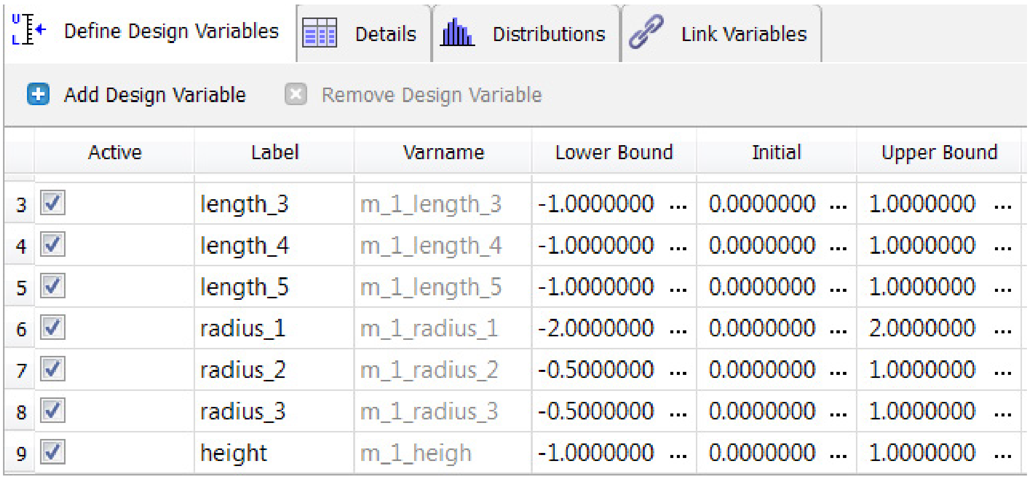




Save the changes as a *.tpl file. The “Define Model” step is concluded by importing the design variables. In the Message Log, the information should be prompted: 9 design variables imported.

7.4.4 Define Design Variables and Nominal Run

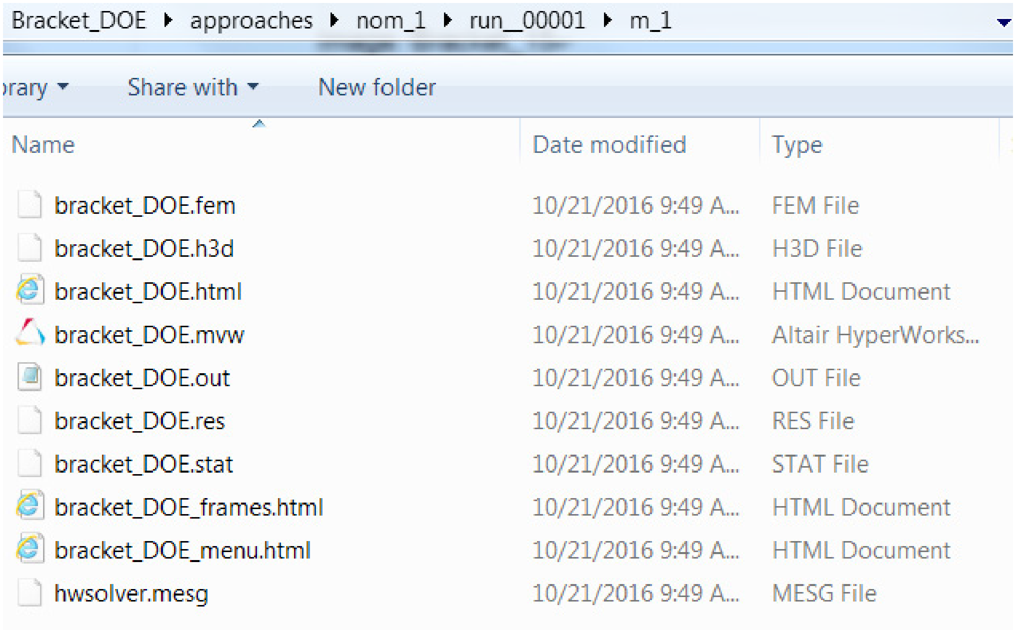
The lower and upper bounds on the design variables can now be modified.



List of the all variables values

Before the Nominal Run (Evaluate Task) is started, make sure that Write, Execute and Extract are active (see below).

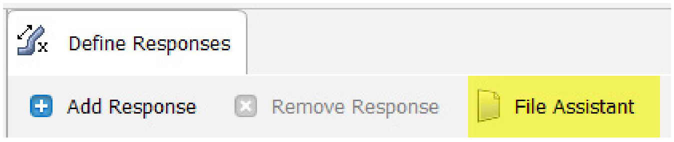
All the analysis files are written to the subdirectory: nom_1/run_00001/m_1



7.4.5 Create Responses

In this study, we want to analyze the volume, the displacement at the force application (node 35527) and the maximum stresses.

Hence, 3 responses will be created.



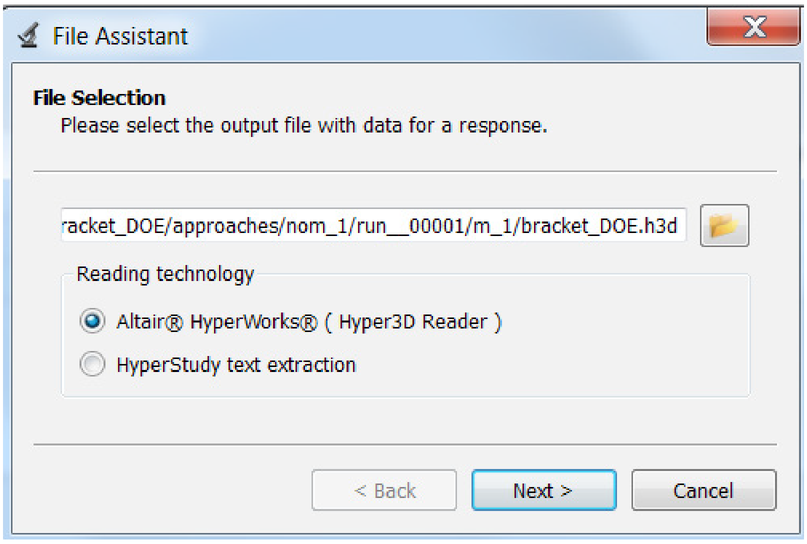
In the previous examples responses were added (created) through "Adding Response". This time however, we are going to employ the "File Assistant" – an easy to use tool/assistant.

In the "File Assistant" you need to specify where the information of interest (=result) is coming from (displacements, stresses etc.

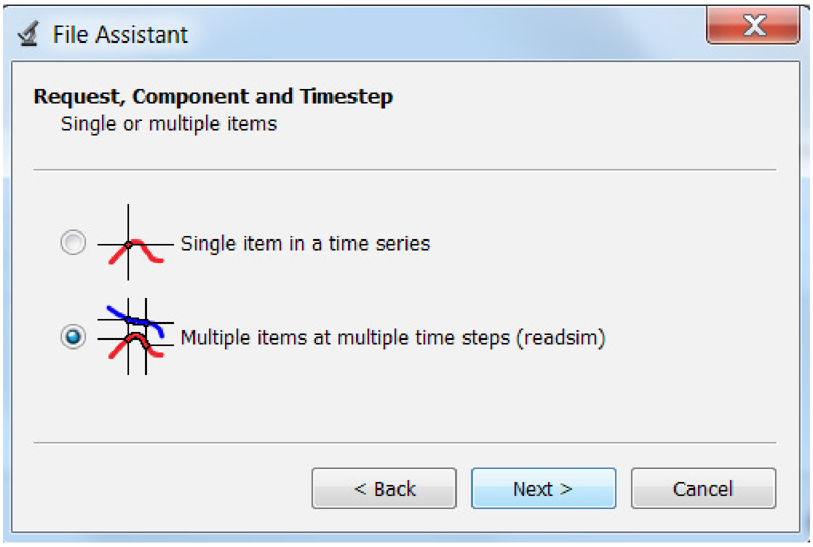
→ *.h3d file; volume, mass etc. → *.out file).

In the File Assistant dialog, set the Reading technology to Altair® HyperWorks® (Hyper3D Reader), then click Next.

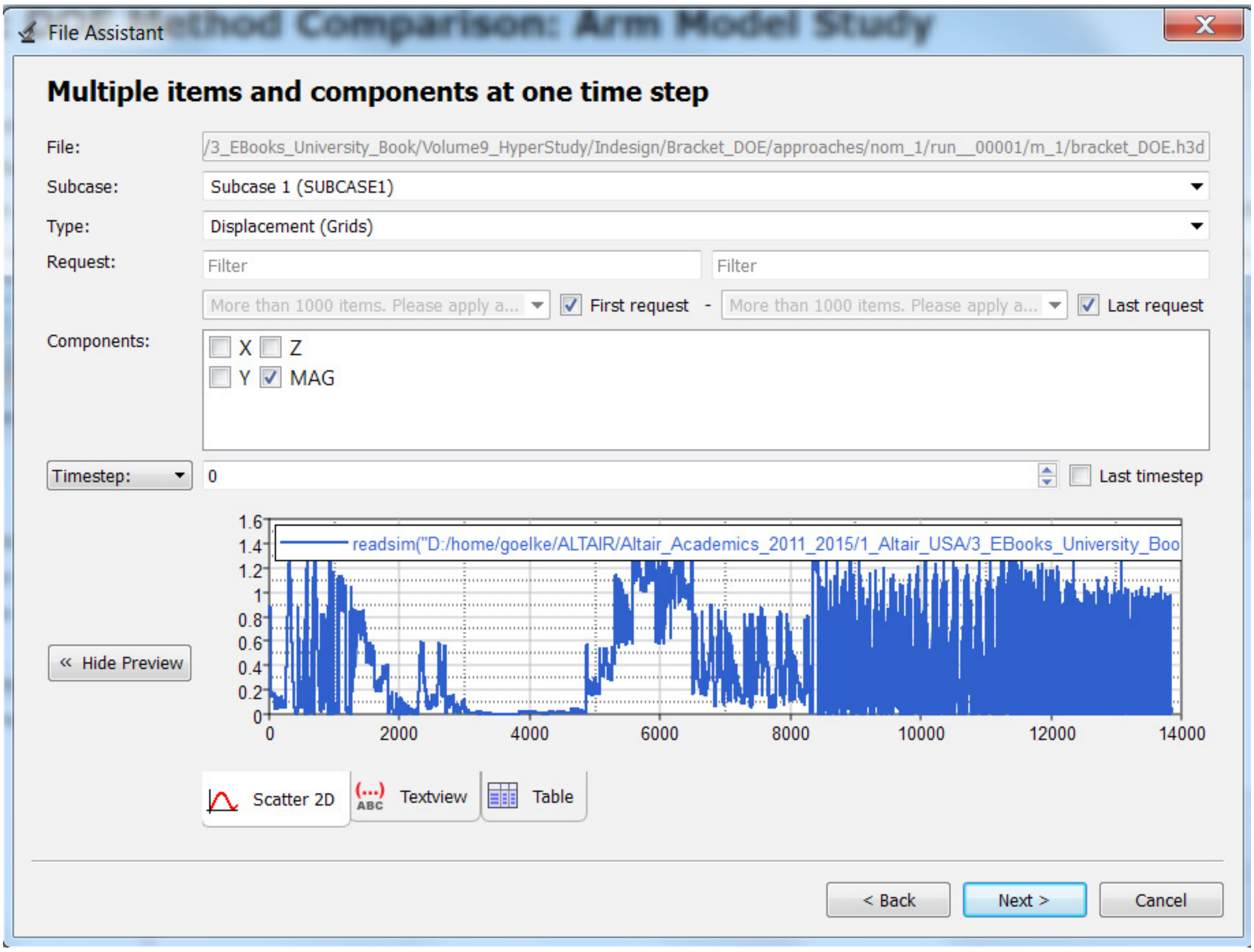
We start defining the Response Displacement.



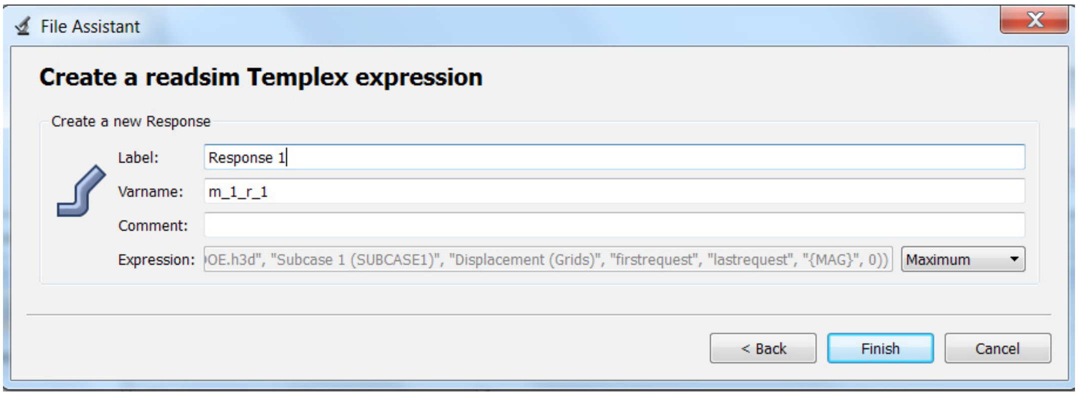
Select “Multiple items at multiple time steps (readsim)”, then Next.



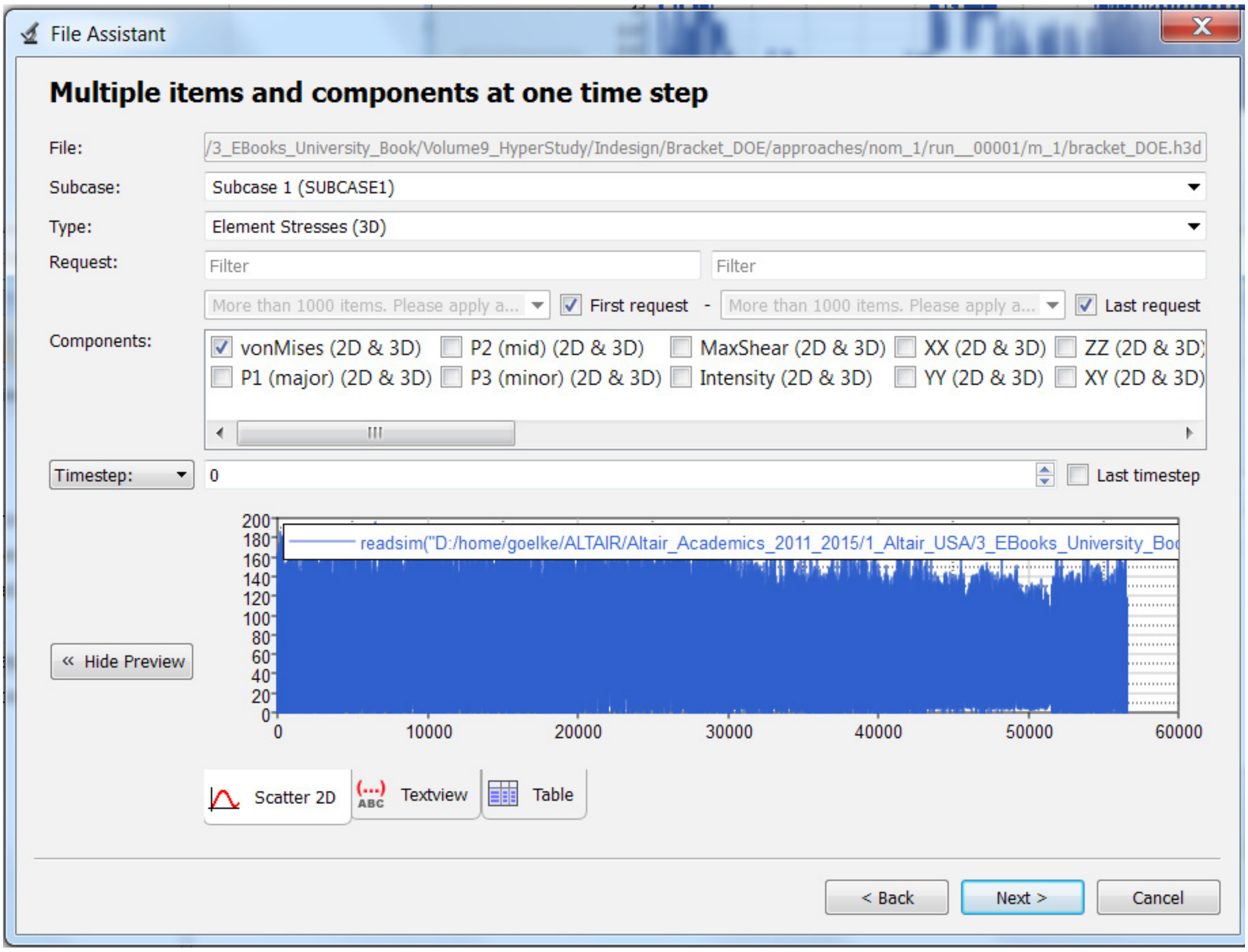
Type is set to “Displacements”, Request is using “First request” to “Last request” (which corresponds to all grids/nodes), Components is MAG(nitude)

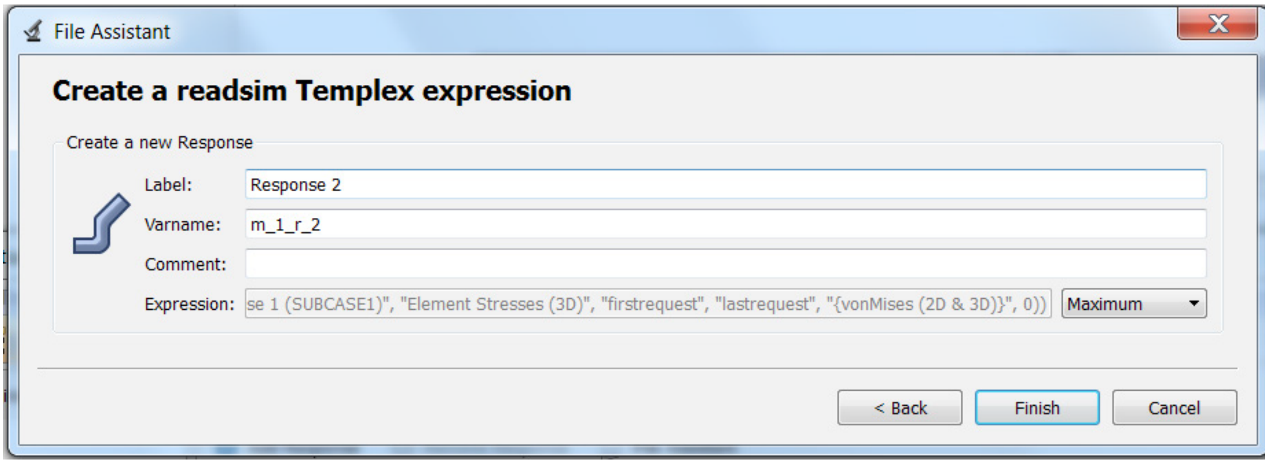


In the above steps the information of interest was specified. Next, we build a response out of the above information. In detail, we are looking at the maximum displacement (magnitude) of all Grids/nodes.

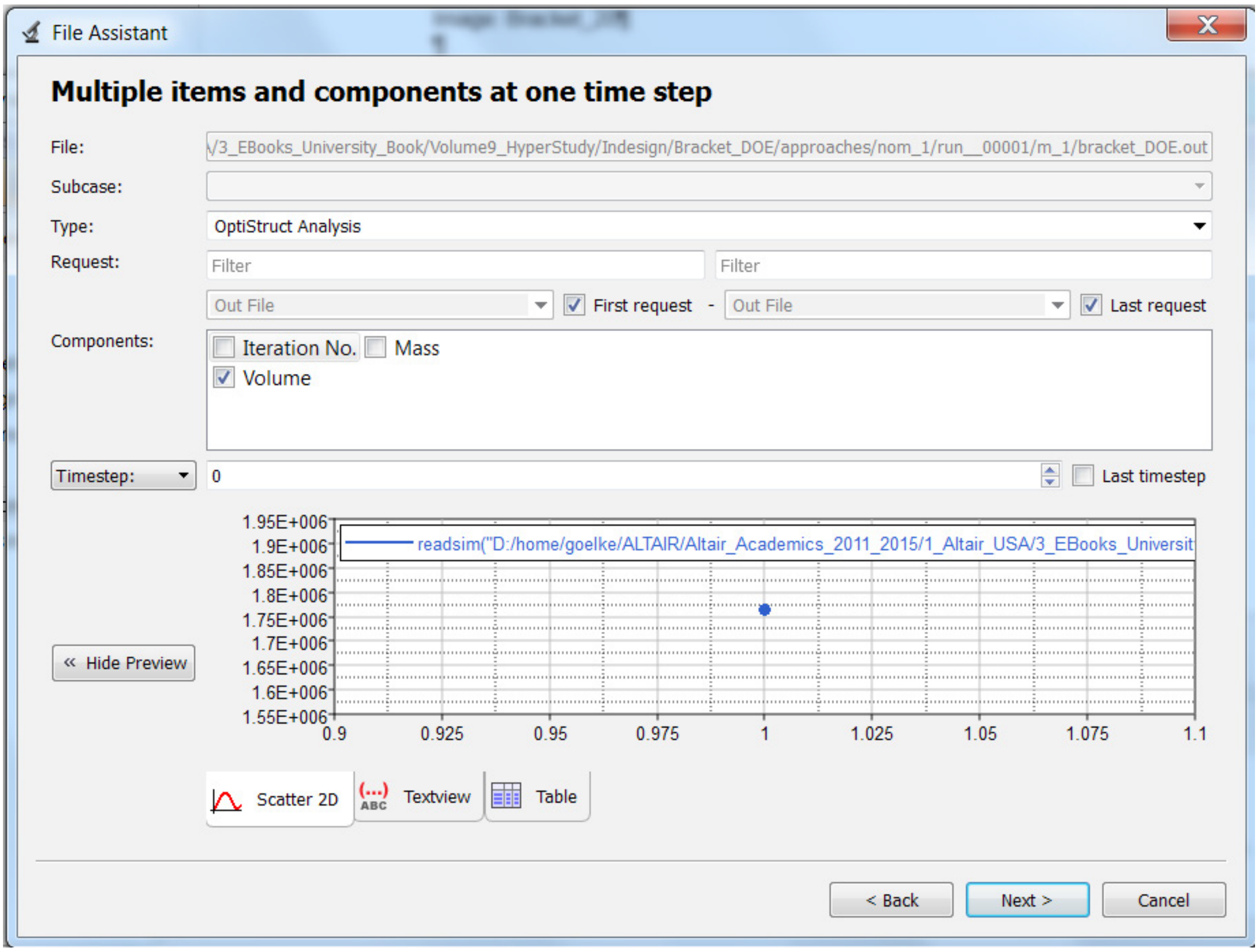


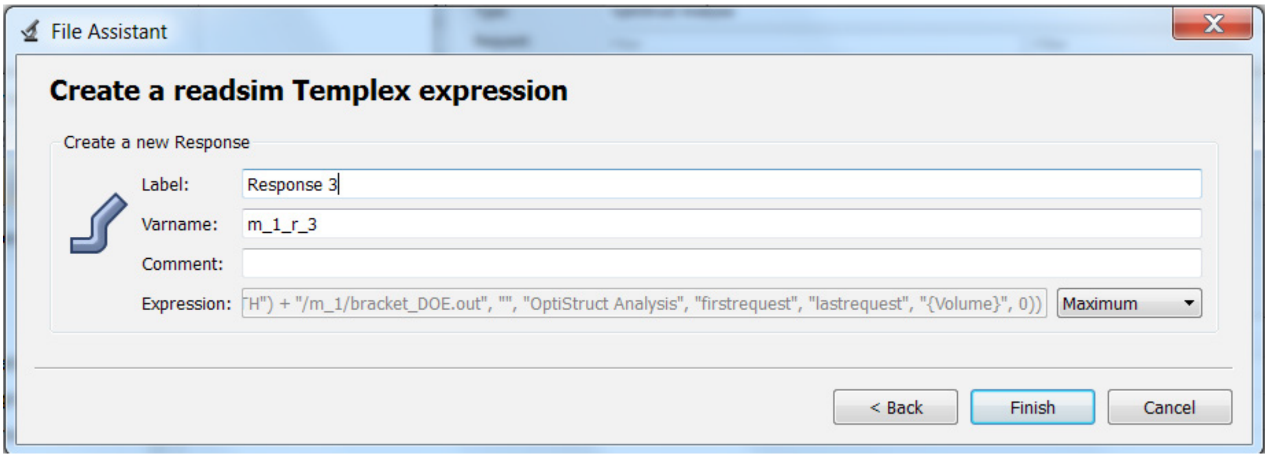
The same steps are repeated regarding the Response "Stress" (von Mises) as depicted in the images below. Again, we are looking at the maximum stress values of all elements.



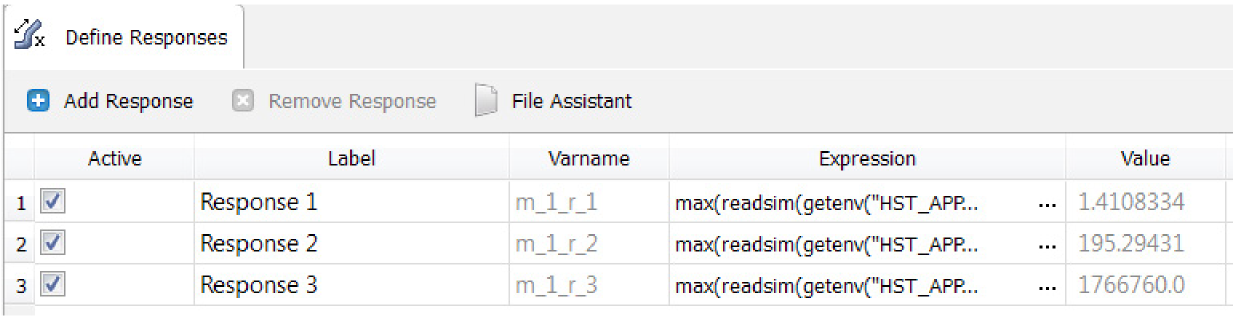


And finally, we define the response of type "Volume".





The Response and their respective values (derived from the nominal run):



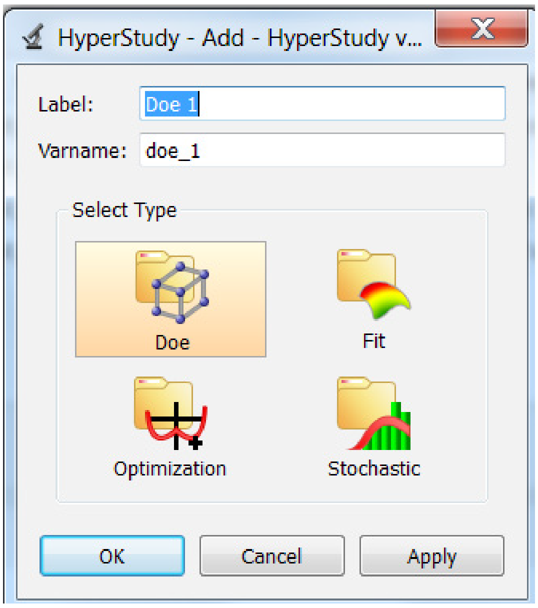
This completes the study setup. You can now proceed to the desired study type (DOE, Optimization, or Stochastic study). Here, we are going to look at the different DOE methods.

7.5 Parameter Screening DOE Setup

In this section, we will study the results from different parameter screening DOE's.

Once the study setup is completed, an Approach (DOE, Optimization, etc.) must be selected.

Right click in the Explorer opens the pop-up window; select "Add Approach" which then depicts the different Approaches. Of course, we select DOE. It is a good practice to use labels which briefly describe the essence of the approach e.g. FR_RES_V (Fractional_Resolution_V)

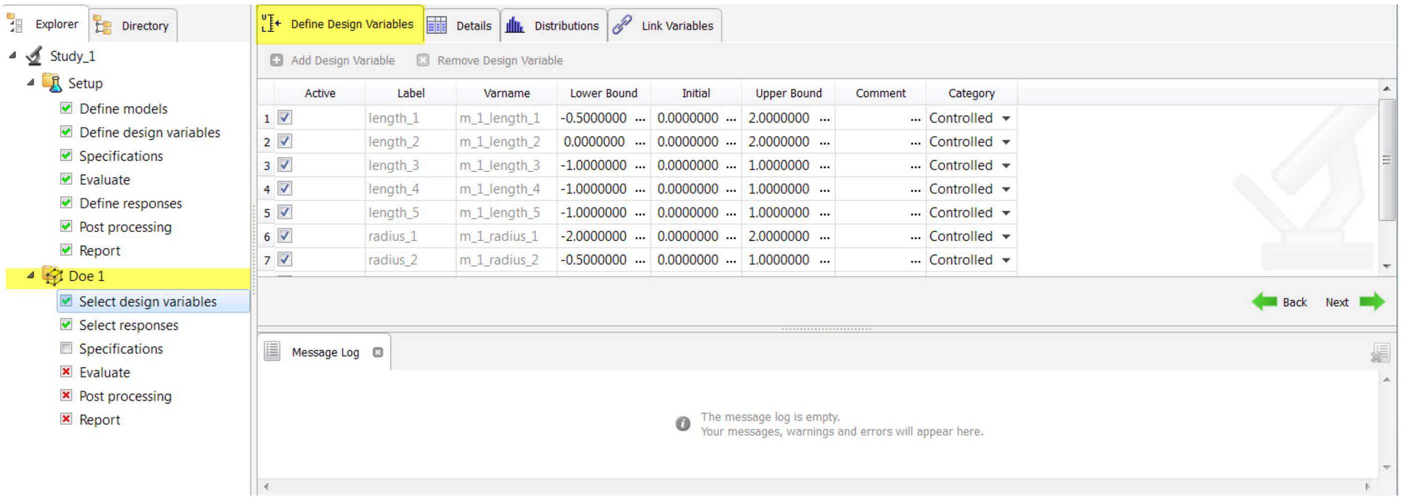


7.5.1 Full Factorial DOE

First one is a Full Factorial DOE. In this study, we will set the shape variables to two levels. Since full factorial runs all combinations of design variable values, we have minimal loss of accuracy, however we have an expensive DOE. We will use this study as our reference to compare the loss of accuracy from other DOE's that are less expensive.

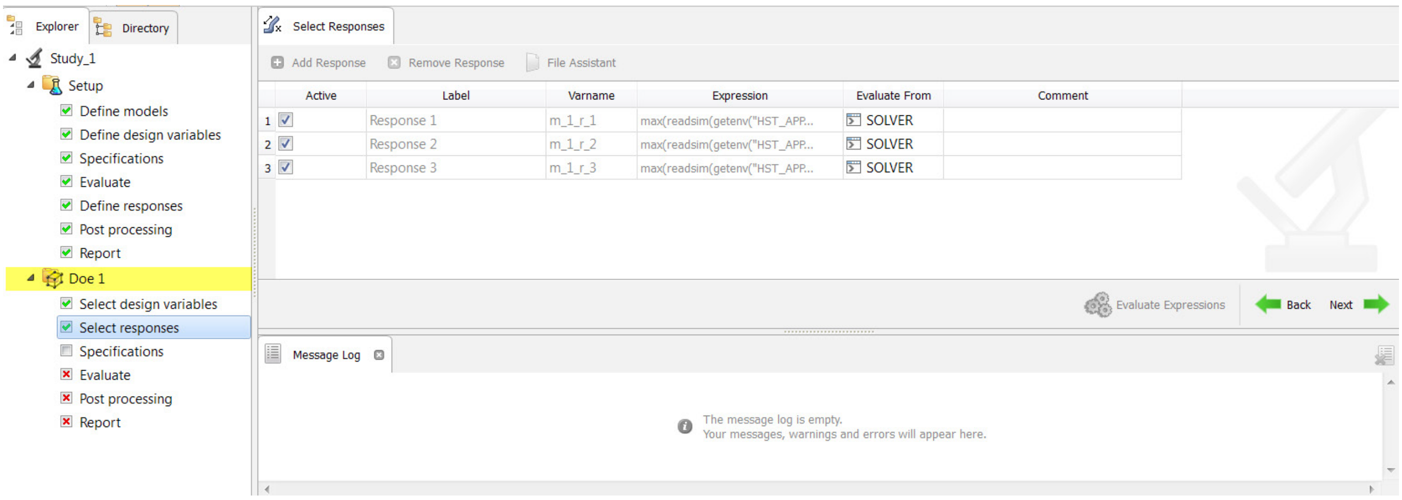
Using 9 shape variables with 2 levels gives a full factorial plan made of 512 runs (LV = 29).

As this is the first DOE we are looking at (from within HyperStudy) we are including some additional information.

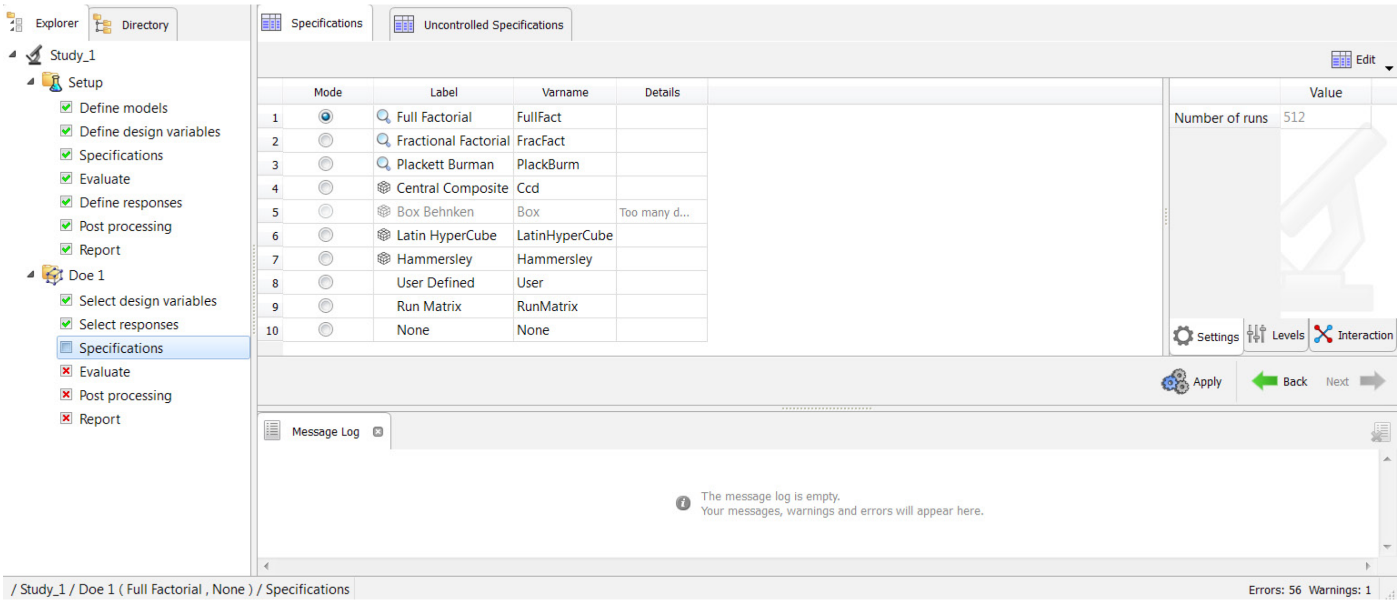


In the "select design variables" step all defined variables are listed. As indicated in the column, all design variables are active currently (but could be made inactive). Also, their bounds are listed again (for review and/or editing purposes).

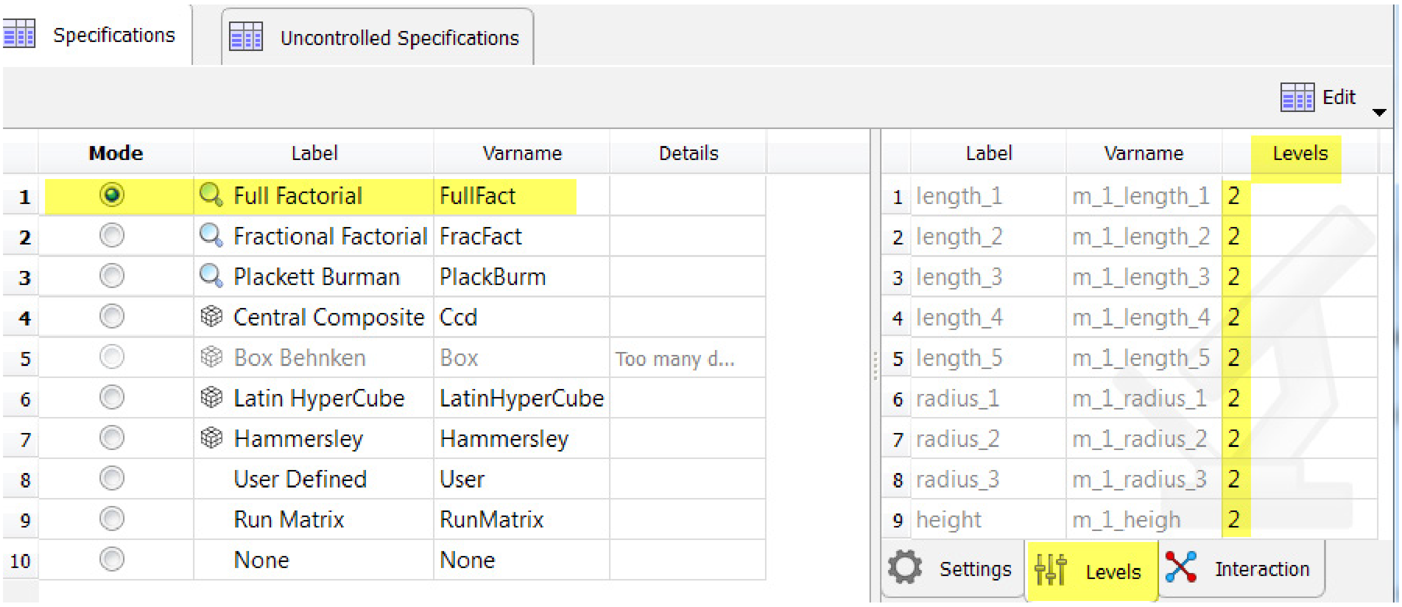
In a similar way all active/inactive responses are listed (but can't be edited in here anymore).



Eventually, in the "Specifications" step the DOE Method to be used is determined (these methods were described earlier).



To check (and edit) the number of Levels (should be two, of course) activate the "Levels" tab. The number of runs associated with the Full Factorial study (512 runs; LV) is depicted on the right.

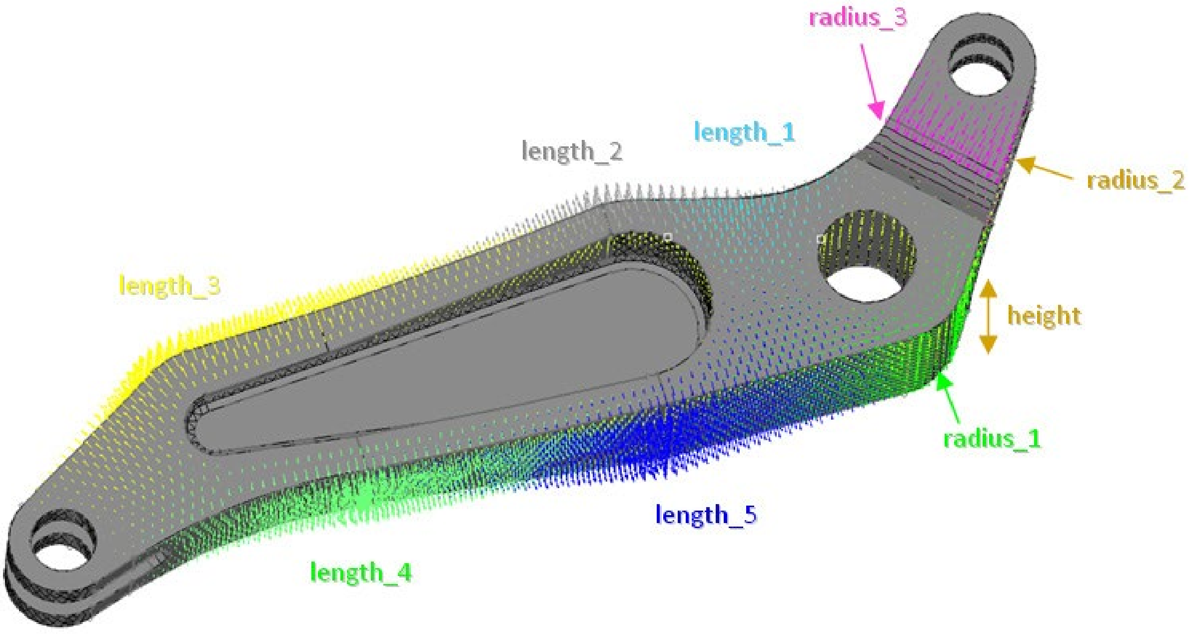


Each design variable is assigned two levels, lower bound and upper bound.

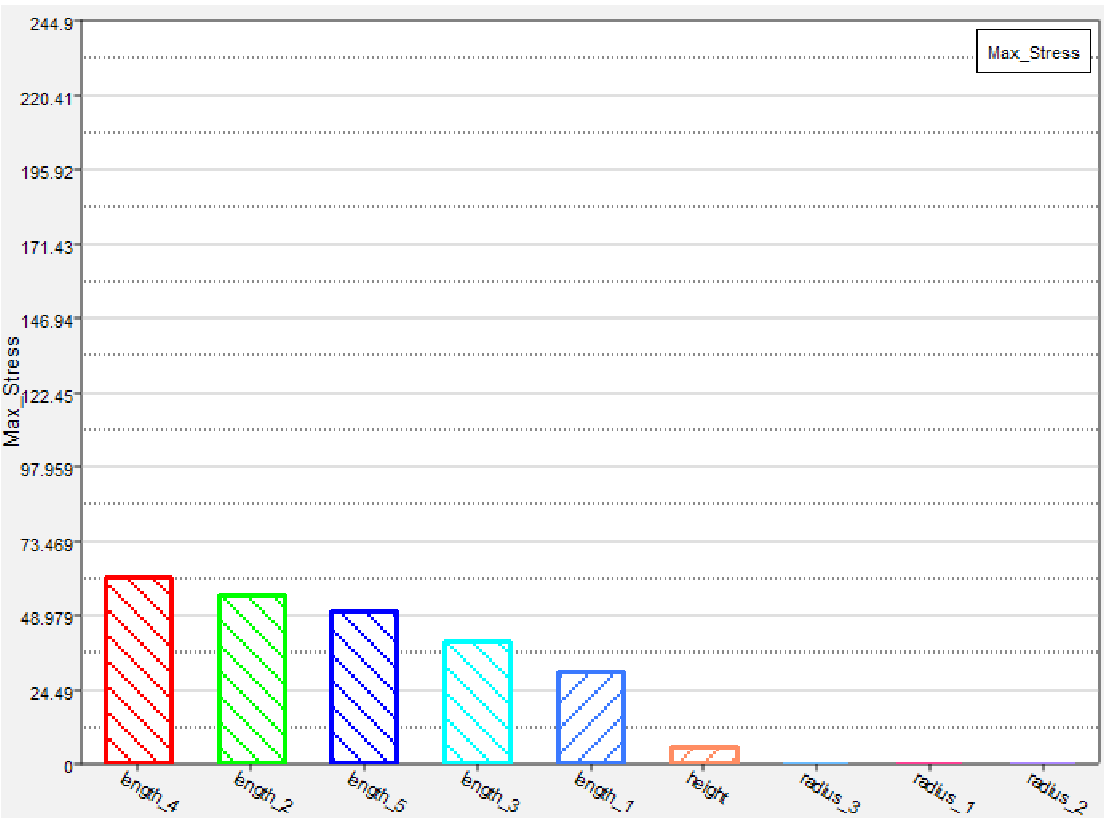
Without getting too trivial: the nominal run lasted about 30 sec on a small laptop. Running a full factorial study would be pretty time consuming already.

For the sake of focus, each screening we will discuss only the results obtained on the Max_stress output response using Pareto plot, Linear Effects and Interactions displays. In an actual study, all responses should be considered in making decisions about parameter screening.

To ease comparison of the graphs with the model (actually with the definitions of length, radius, height) we are showing the initial model again.

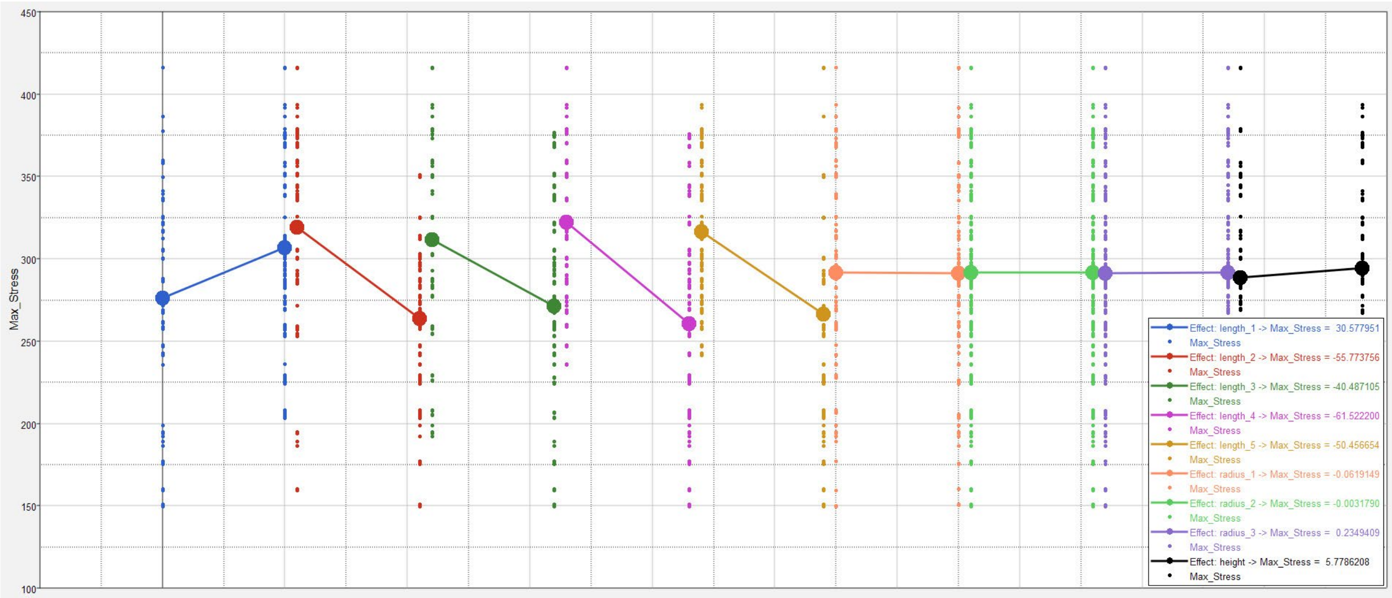


Shapes defined on the "arm model"



A Pareto Plot shows the ranked influence (highest to lowest) of the design variables on the response.

We can see that the 5 lengths have the largest influence and the 3 radius have the least influence on Max_stress. The hashed lines slope (positive or negative) indicates a positive or a negative effect of a variable on the response. In this case length_4, length_2, length_5 and length_3 have negative slopes which means that if these variables increase Max_stress will decrease. On the other hand, length_1 has a positive slope which means that increasing length_1 increases Max_stress.

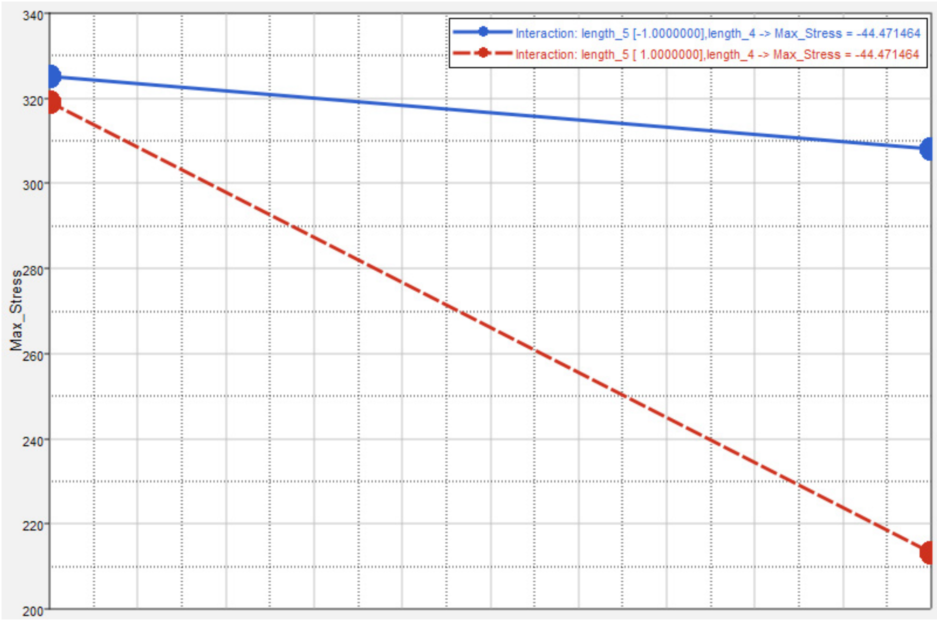


Linear effect – max stress

From the graph above it can be concluded that:

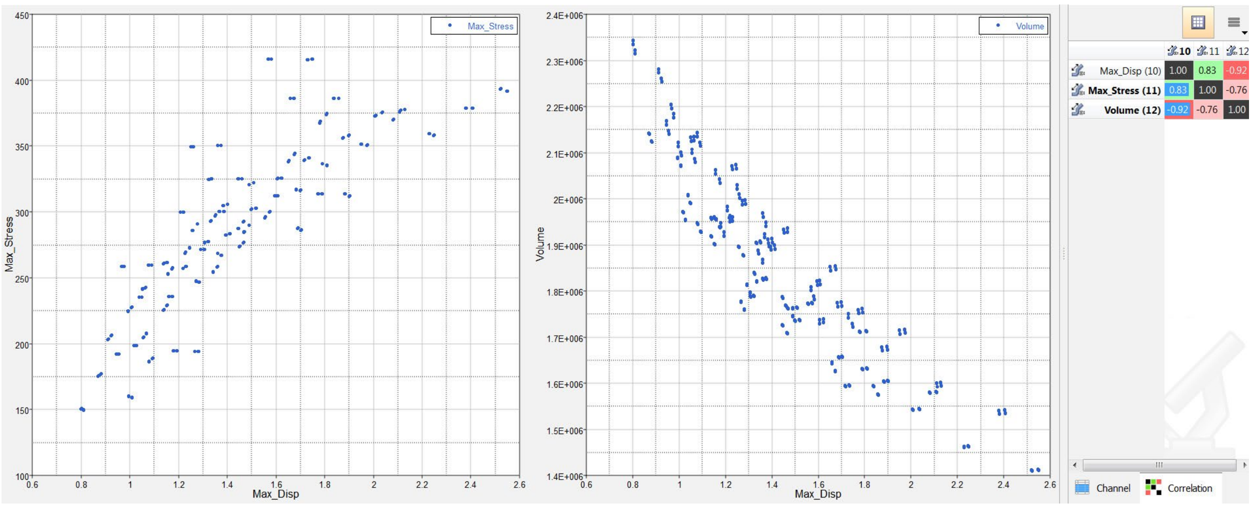
- the three radii (plotted on the right) have almost no effect on the responses
- length 2, 3, 4 and 5 have the same type of effect (direction and magnitude)
- length 1 and height have the same direction of effect but are less important than the four other length variables

In addition to the linear effects, we can look into interactions between design variables. Looking at the Max_stress response, you can see that a number of interactions are very small. You can still find cases with a true interaction, such as the interaction of length 4 and length 5 illustrated in the image below. The effect of variable length 4 on response Max_Stress is in the same direction whatever the value of length 5 is, but the effect (magnitude) is much more important when length 5 is large.



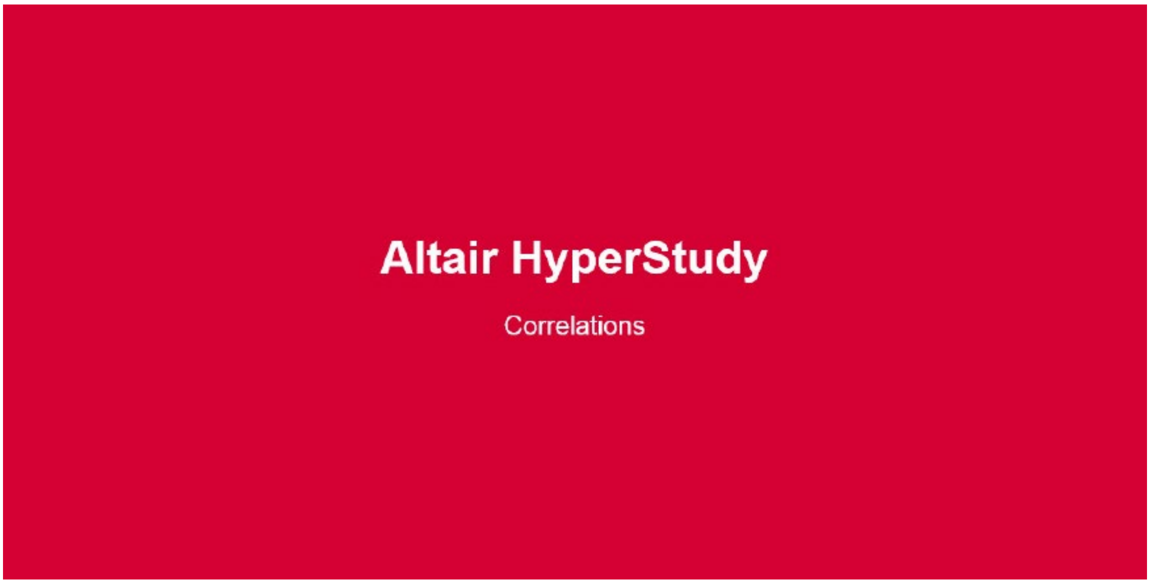
Length 4 and length 5 on max_stress

The scatter plot indicates a strong negative correlation between the volume of the part and its structural responses.



Stress vs displacement and displacement vs volume

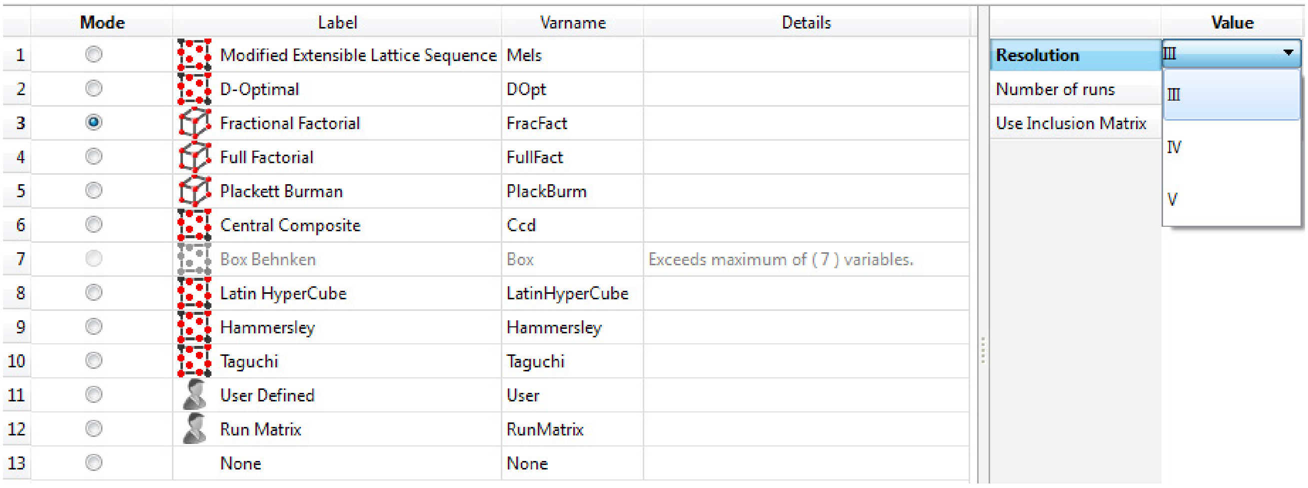
In the video below Fatma Koçer provides some more information about “Scatter Plots”.



https://altair-2.wistia.com/medias/x9xofxd9o

7.5.2 Resolution III, IV and V Fractional Factorial DOE with 2 Levels

Next, we set up a Fractional Factorial DOE with the same levels as the previous full factorial DOE. Using the 9 design variables with 2 levels, without toggling ON any interaction, leads to a 12 runs for a Resolution III Fractional factorial plan. Repeat the steps for Resolution IV and V to observe the differences between accuracy and effort between these three resolutions.



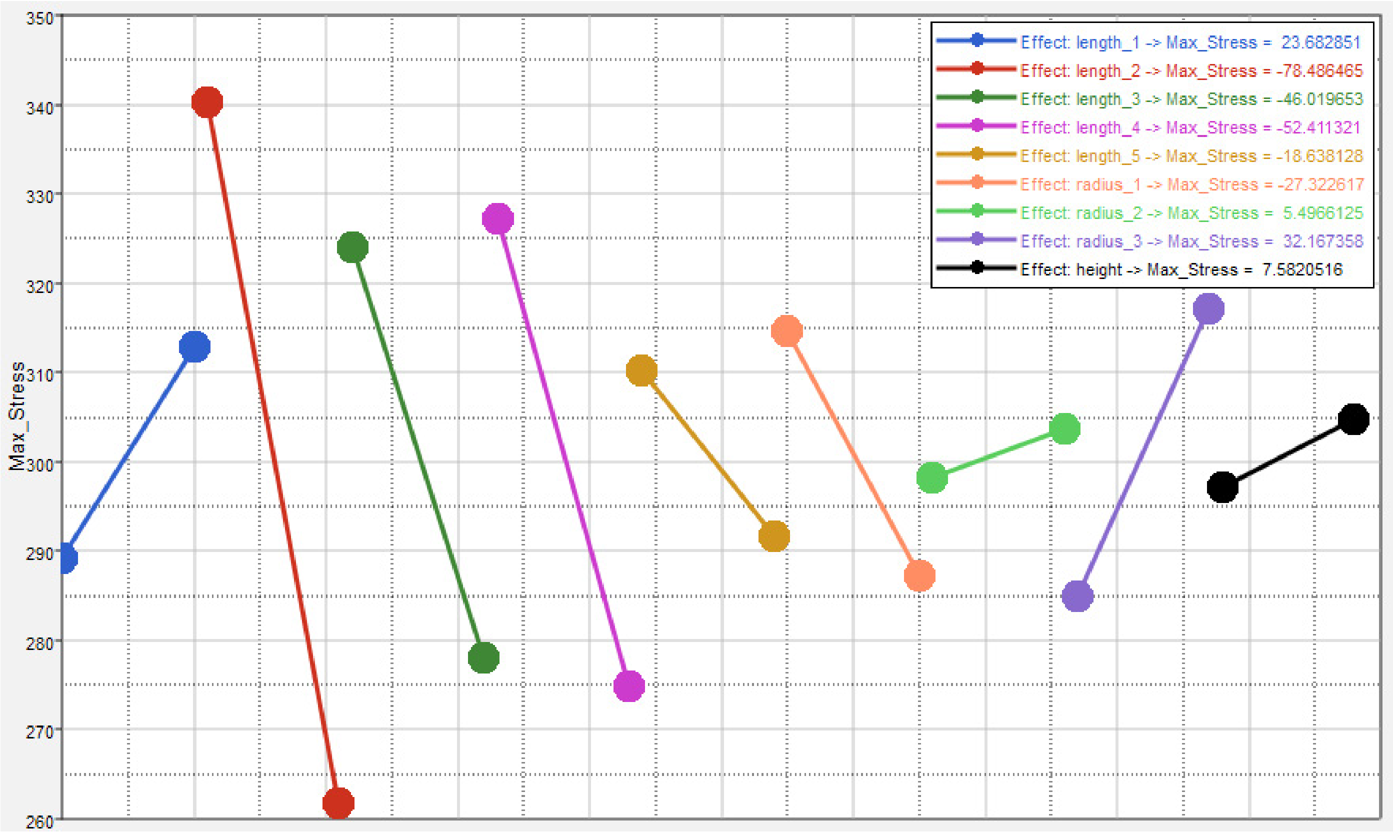
	Mode	Label	Varname	Details
1		Modified Extensible Lattice Sequence	Mels	
2		D-Optimal	DOpt	
3	●	Fractional Factorial	FracFact	
4		Full Factorial	FullFact	
5		Plackett Burman	PlackBurm	
6		Central Composite	Ccd	
7		Box Behnken	Box	Exceeds maximum of (7) variables.
8		Latin HyperCube	LatinHyperCube	
9		Hammersley	Hammersley	
10		Taguchi	Taguchi	
11		User Defined	User	
12		Run Matrix	RunMatrix	
13		None	None	

What do we learn from the graphs below?

From the graphs below, it is understood that the four length variables (2, 3 4 and 5) have a major effect on the model responses.

The three radii variables have a negligible effect on the volume and a small to medium effect on the displacement and stress.

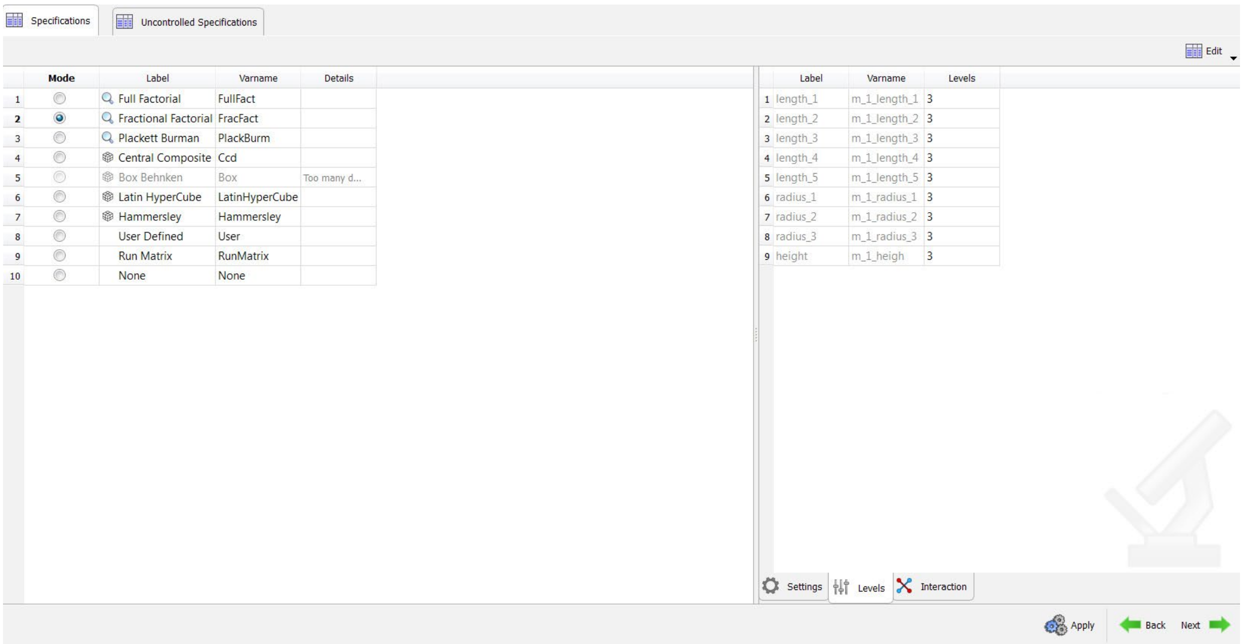
It is also understood that length_1 have an opposite influence on the responses compared to the other length shapes. It is seen that length and height have significant influences and we can focus on those variables for the rest of the studies. As for the three radii, we will set them to their nominal values to get the starting point.



Main effect stress

7.5.3 Fractional Factorial DOE with 3 Levels

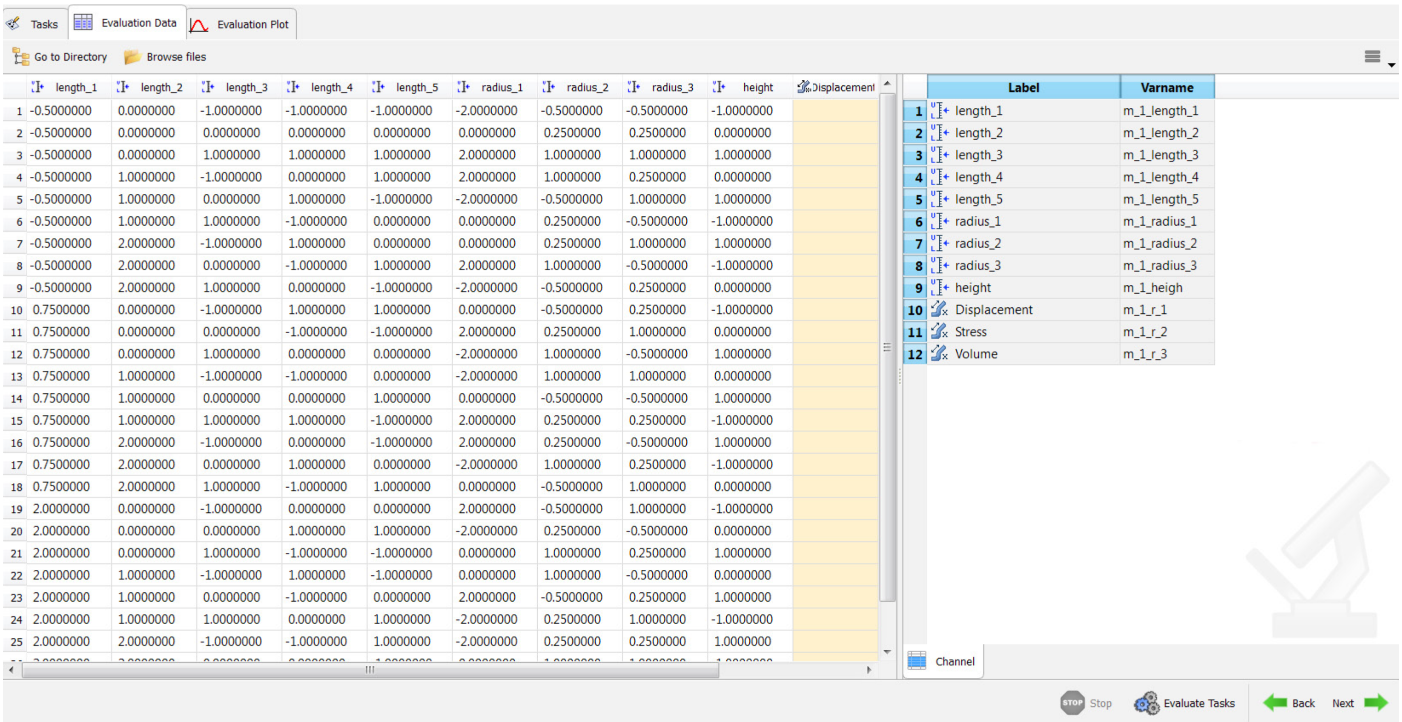
Next, set up a Fractional Factorial DOE with three levels. Using the 9 design variables with 3 levels (resolution V), without toggling ON any interaction, leads to a 27 runs fractional factorial plan.



	Mode	Label	Varname	Details
1		Full Factorial	FullFact	
2	●	Fractional Factorial	FracFact	
3		Plackett Burman	PlackBurm	
4		Central Composite	Ccd	
5		Box Behnken	Box	Too many d...
6		Latin HyperCube	LatinHyperCube	
7		Hammersley	Hammersley	
8		User Defined	User	
9		Run Matrix	RunMatrix	
10		None	None	

	Label	Varname	Levels
1	length_1	m_1_length_1	3
2	length_2	m_1_length_2	3
3	length_3	m_1_length_3	3
4	length_4	m_1_length_4	3
5	length_5	m_1_length_5	3
6	radius_1	m_1_radius_1	3
7	radius_2	m_1_radius_2	3
8	radius_3	m_1_radius_3	3
9	height	m_1_heigh	3

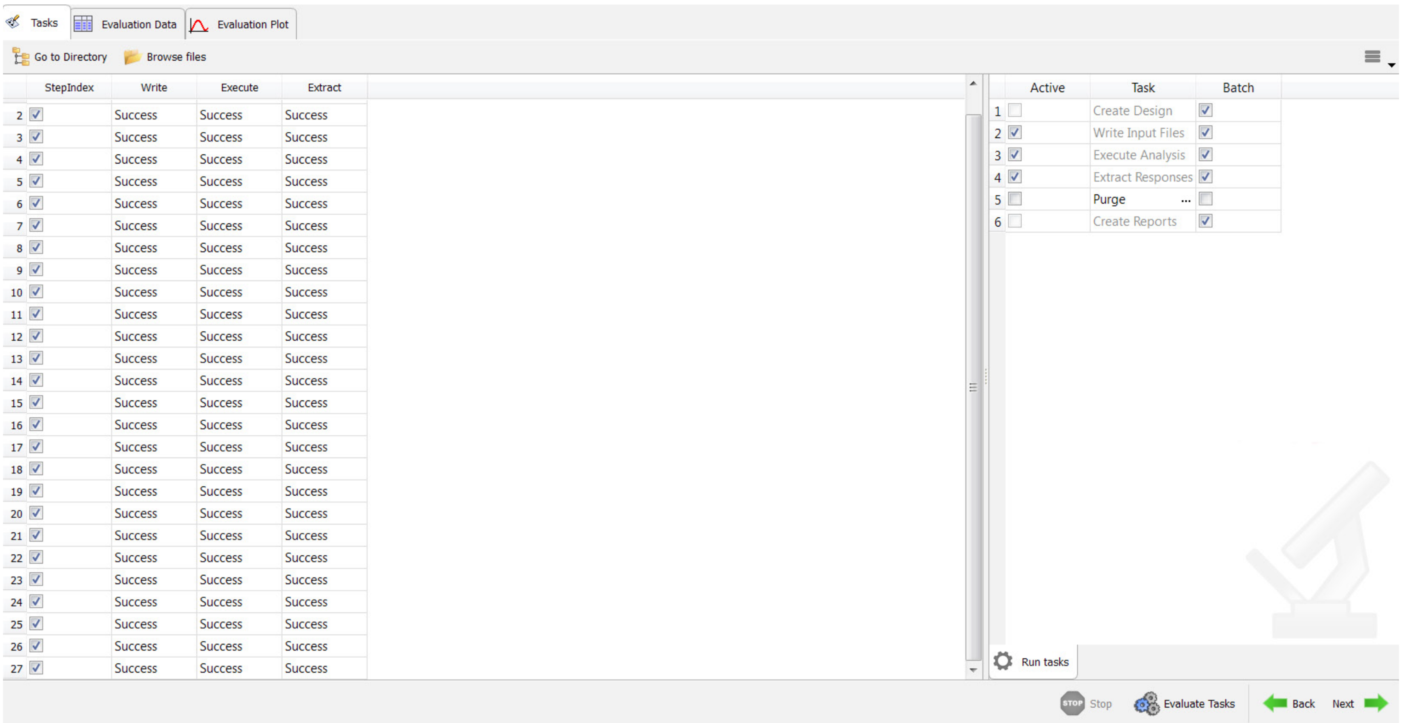
As shown below, the levels are upper, lower and mean value. The initial value of the design variable is not considered (except of course when mean = initial).



	length_1	length_2	length_3	length_4	length_5	radius_1	radius_2	radius_3	height	Displacement
1	-0.5000000	0.0000000	-1.0000000	-1.0000000	-1.0000000	-2.0000000	-0.5000000	-0.5000000	-1.0000000	
2	-0.5000000	0.0000000	0.0000000	0.0000000	0.0000000	0.0000000	0.2500000	0.2500000	0.0000000	
3	-0.5000000	0.0000000	1.0000000	1.0000000	1.0000000	2.0000000	1.0000000	1.0000000	1.0000000	
4	-0.5000000	1.0000000	-1.0000000	0.0000000	1.0000000	2.0000000	1.0000000	0.2500000	0.0000000	
5	-0.5000000	1.0000000	0.0000000	1.0000000	-1.0000000	-2.0000000	-0.5000000	1.0000000	1.0000000	
6	-0.5000000	1.0000000	1.0000000	-1.0000000	0.0000000	0.0000000	0.2500000	-0.5000000	-1.0000000	
7	-0.5000000	2.0000000	-1.0000000	1.0000000	0.0000000	0.0000000	0.2500000	1.0000000	1.0000000	
8	-0.5000000	2.0000000	0.0000000	-1.0000000	1.0000000	2.0000000	1.0000000	-0.5000000	-1.0000000	
9	-0.5000000	2.0000000	1.0000000	0.0000000	-1.0000000	-2.0000000	-0.5000000	0.2500000	0.0000000	
10	0.7500000	0.0000000	-1.0000000	1.0000000	1.0000000	0.0000000	-0.5000000	0.2500000	-1.0000000	
11	0.7500000	0.0000000	0.0000000	-1.0000000	-1.0000000	2.0000000	0.2500000	1.0000000	0.0000000	
12	0.7500000	0.0000000	1.0000000	0.0000000	0.0000000	-2.0000000	1.0000000	-0.5000000	1.0000000	
13	0.7500000	1.0000000	-1.0000000	-1.0000000	0.0000000	-2.0000000	1.0000000	1.0000000	0.0000000	
14	0.7500000	1.0000000	0.0000000	0.0000000	1.0000000	0.0000000	-0.5000000	-0.5000000	1.0000000	
15	0.7500000	1.0000000	1.0000000	1.0000000	-1.0000000	2.0000000	0.2500000	0.2500000	-1.0000000	
16	0.7500000	2.0000000	-1.0000000	0.0000000	-1.0000000	2.0000000	0.2500000	-0.5000000	1.0000000	
17	0.7500000	2.0000000	0.0000000	1.0000000	0.0000000	-2.0000000	1.0000000	0.2500000	-1.0000000	
18	0.7500000	2.0000000	1.0000000	-1.0000000	1.0000000	0.0000000	-0.5000000	1.0000000	0.0000000	
19	2.0000000	0.0000000	-1.0000000	0.0000000	0.0000000	2.0000000	-0.5000000	1.0000000	-1.0000000	
20	2.0000000	0.0000000	0.0000000	1.0000000	1.0000000	-2.0000000	0.2500000	-0.5000000	0.0000000	
21	2.0000000	0.0000000	1.0000000	-1.0000000	-1.0000000	0.0000000	1.0000000	0.2500000	1.0000000	
22	2.0000000	1.0000000	-1.0000000	1.0000000	-1.0000000	0.0000000	1.0000000	-0.5000000	0.0000000	
23	2.0000000	1.0000000	0.0000000	-1.0000000	0.0000000	2.0000000	-0.5000000	0.2500000	1.0000000	
24	2.0000000	1.0000000	1.0000000	0.0000000	1.0000000	-2.0000000	0.2500000	1.0000000	-1.0000000	
25	2.0000000	2.0000000	-1.0000000	-1.0000000	1.0000000	-2.0000000	0.2500000	0.2500000	1.0000000	

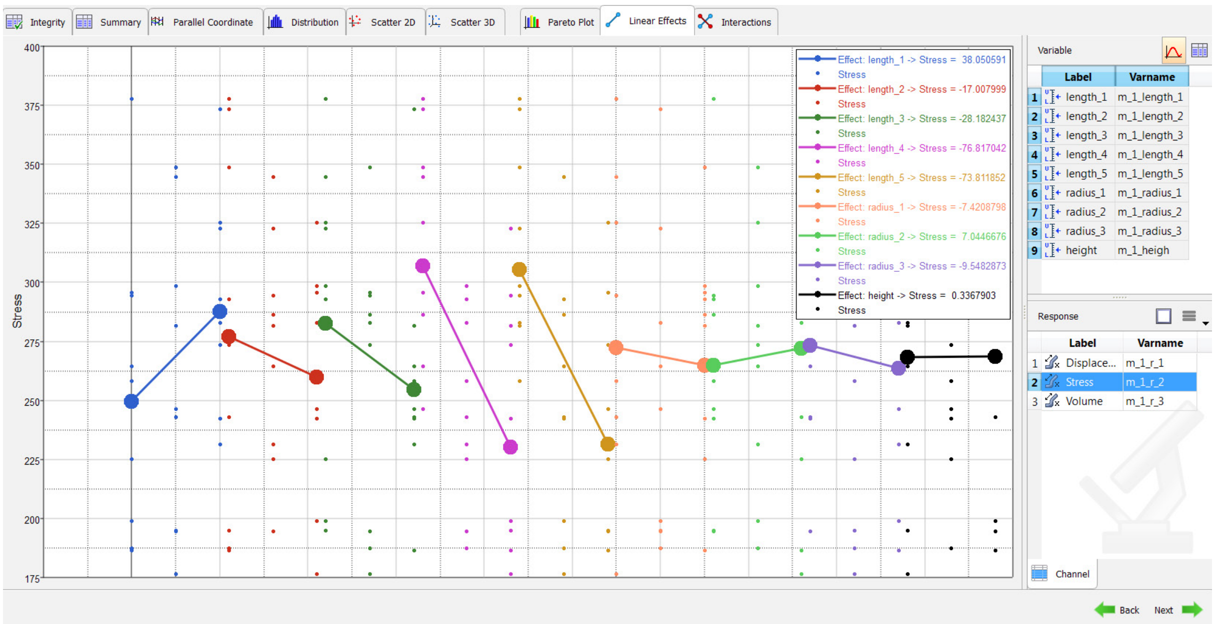
	Label	Varname
1	length_1	m_1_length_1
2	length_2	m_1_length_2
3	length_3	m_1_length_3
4	length_4	m_1_length_4
5	length_5	m_1_length_5
6	radius_1	m_1_radius_1
7	radius_2	m_1_radius_2
8	radius_3	m_1_radius_3
9	height	m_1_heigh
10	Displacement	m_1_r_1
11	Stress	m_1_r_2
12	Volume	m_1_r_3

Levels for the 27 runs of the Fractional Factorial DOE

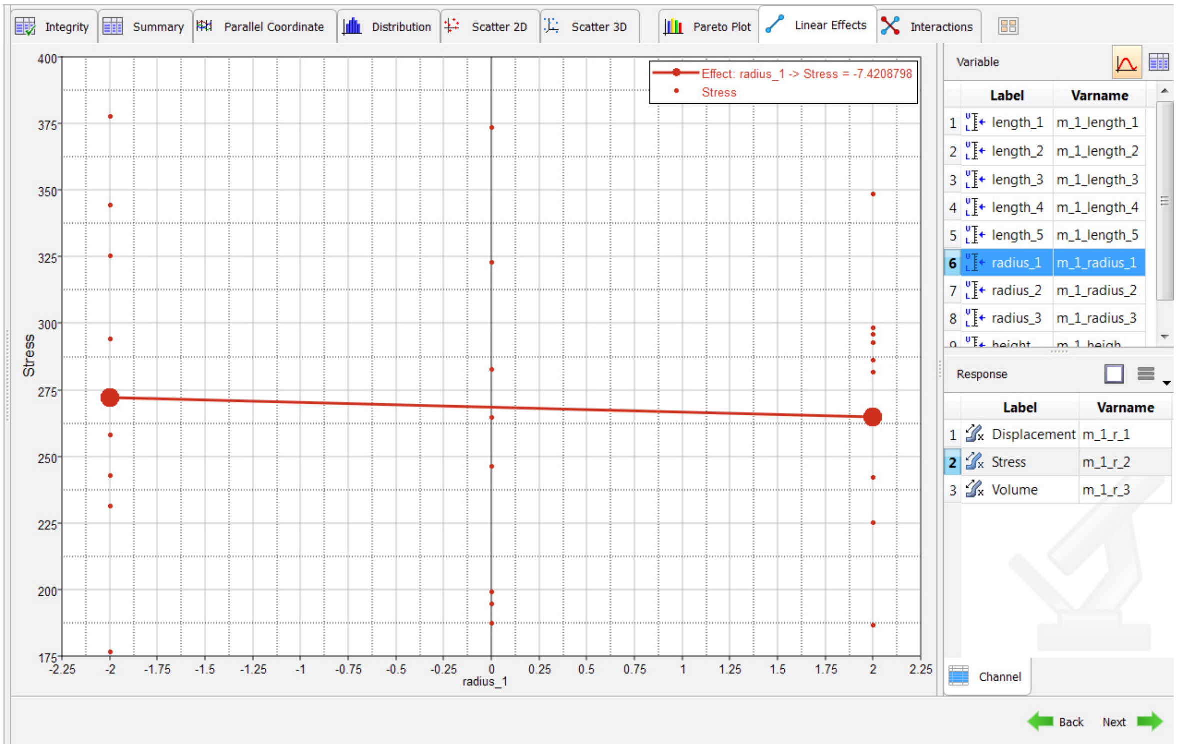


The main effects plots displayed in the figures below give the following indications:

- the 3 radii variables have small effects
- length 1 is opposed to the other length variables



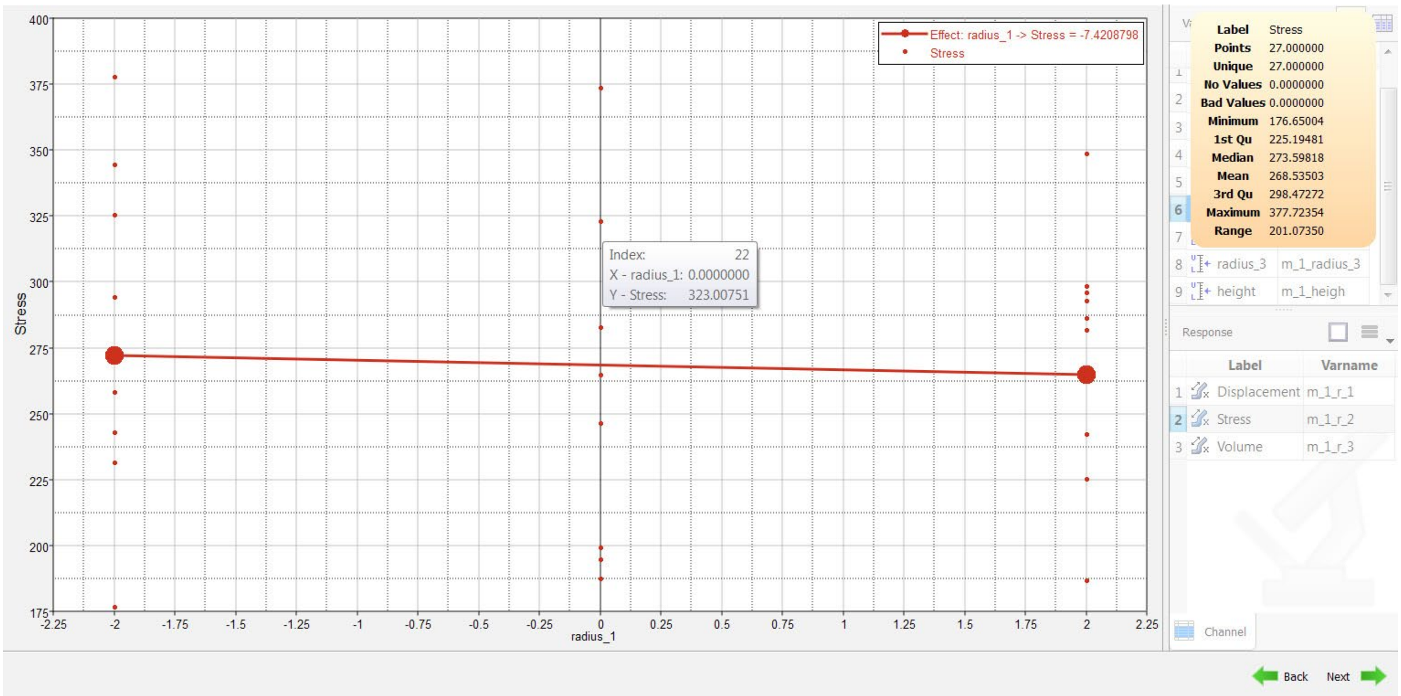
Stress



Effect of radius 1 on stress

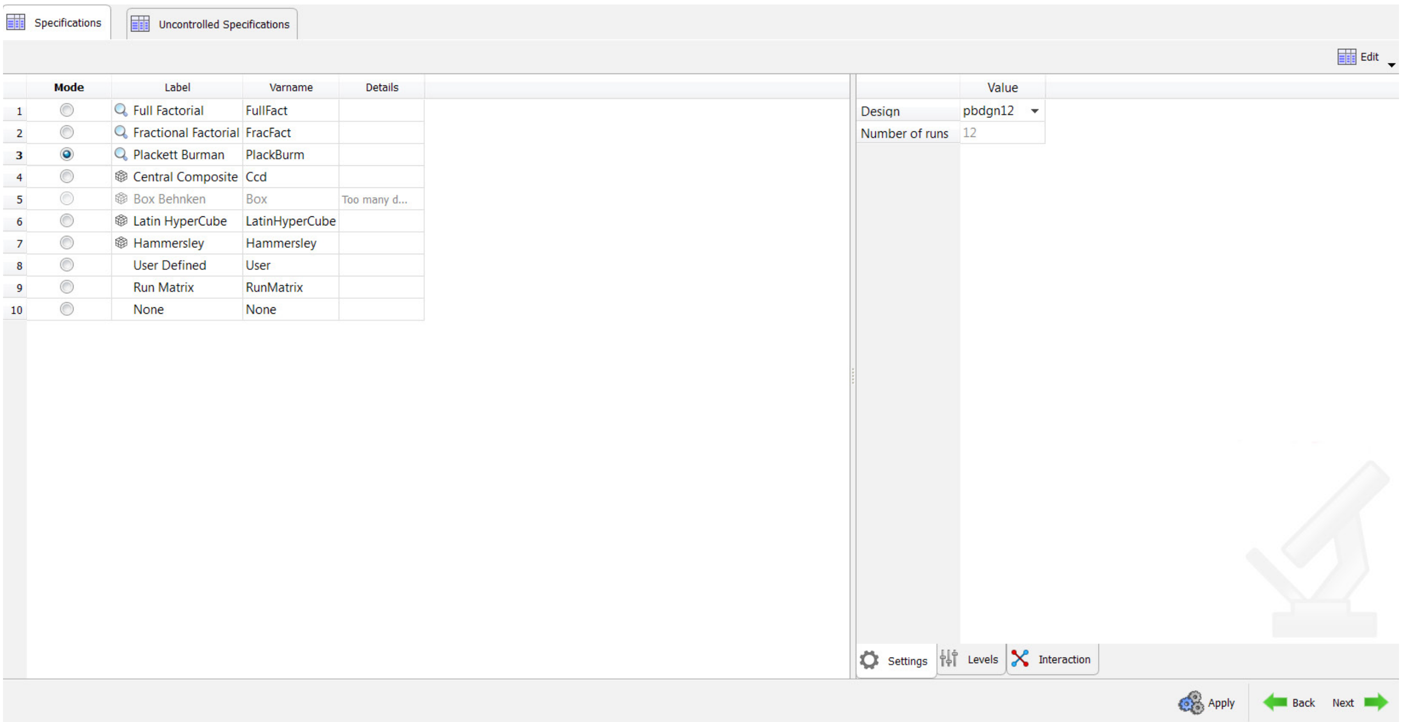
Kind of reminder: In the Linear Effects Plot the averaged results (for two level studies) at lower and upper bounds of the design variable are "connected" through a straight line. Looking at the image above non-linear effects cannot be observed.

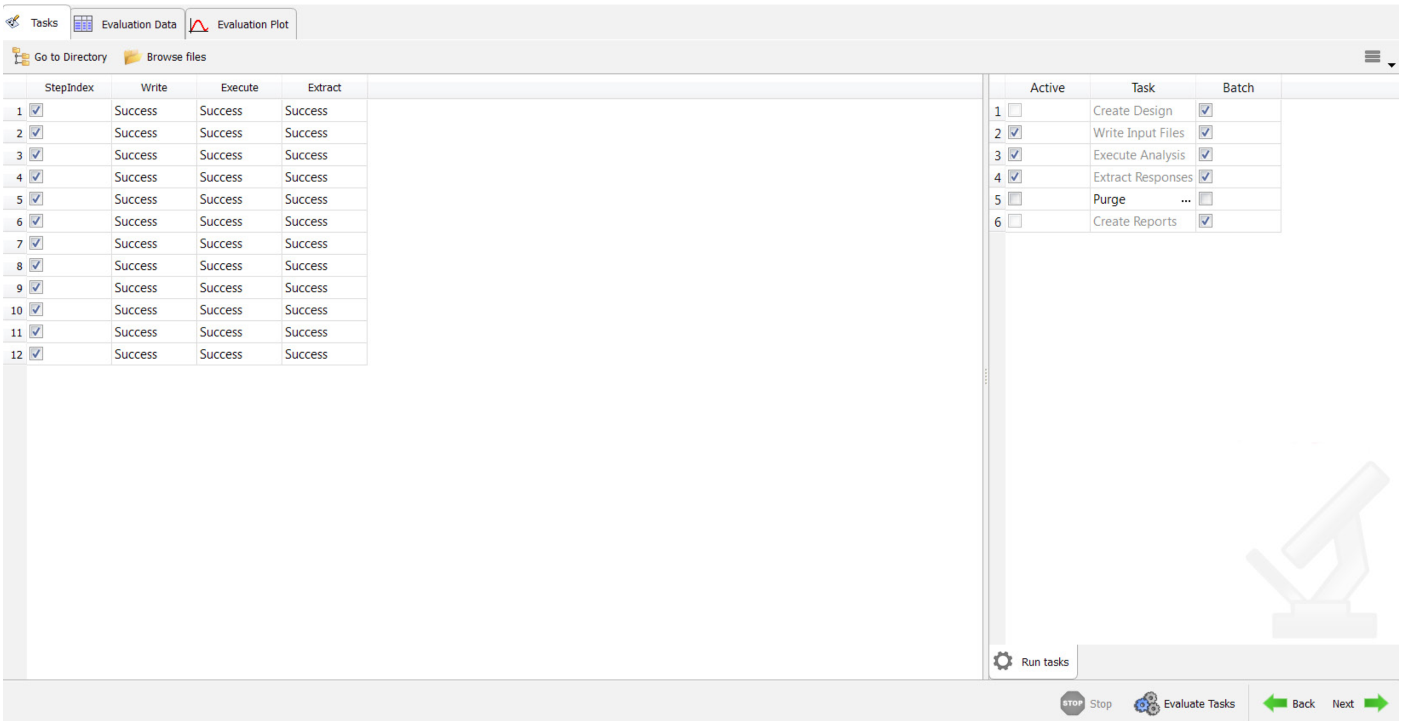
BTW, by hovering the mouse cursor over the data point you can extract information about the run number, the radius_1 value and the associated response "stress" (depicted in the small pop-up window). The pop-up window to the right provides detailed information about the "radius_1 – Stress" experiment, i.e. 27 runs, mean value 268,53 etc.

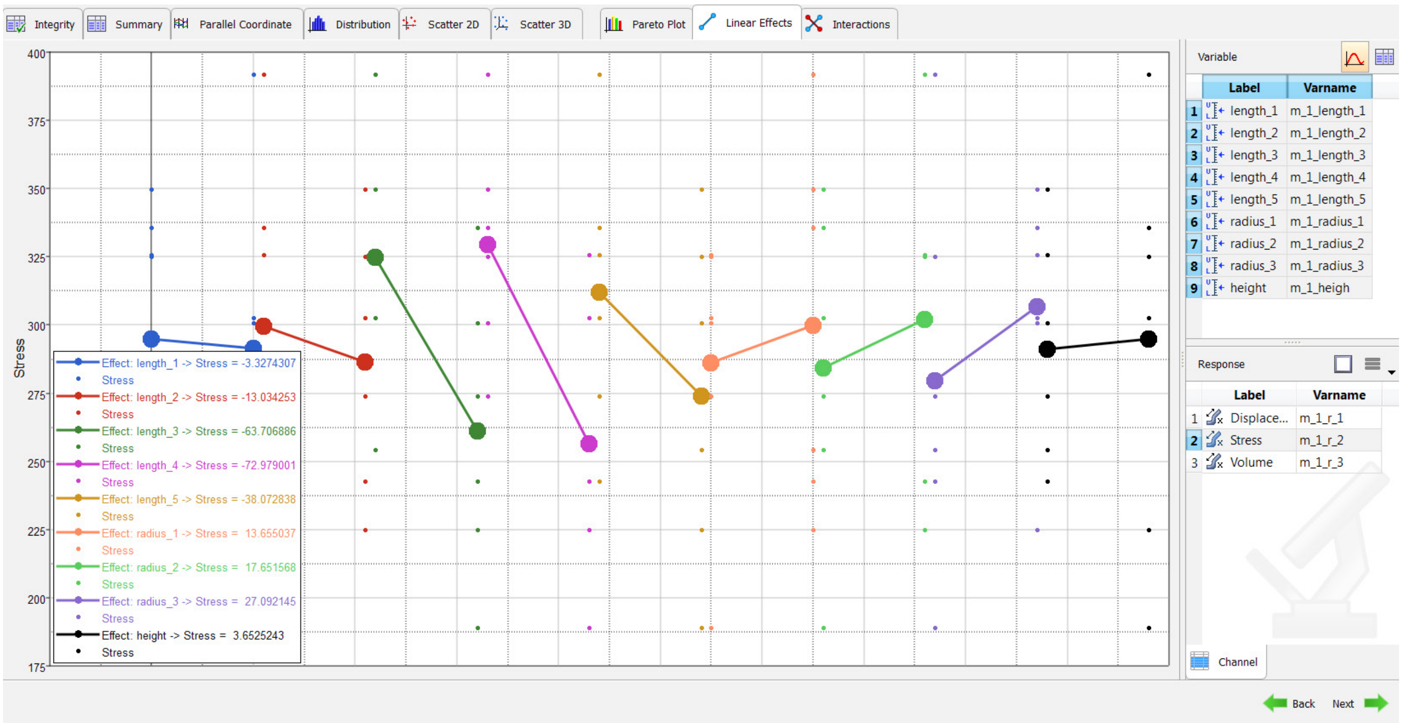


7.5.4 Plackett-Burman DOE

Next, set up a Plackett-Burman DOE which uses 2 levels for all the design variables. For this study, using 9 variables, we get a 12-run matrix.







Stress

Note: The "Scatter" can be turned off if wanted (see inset in image)

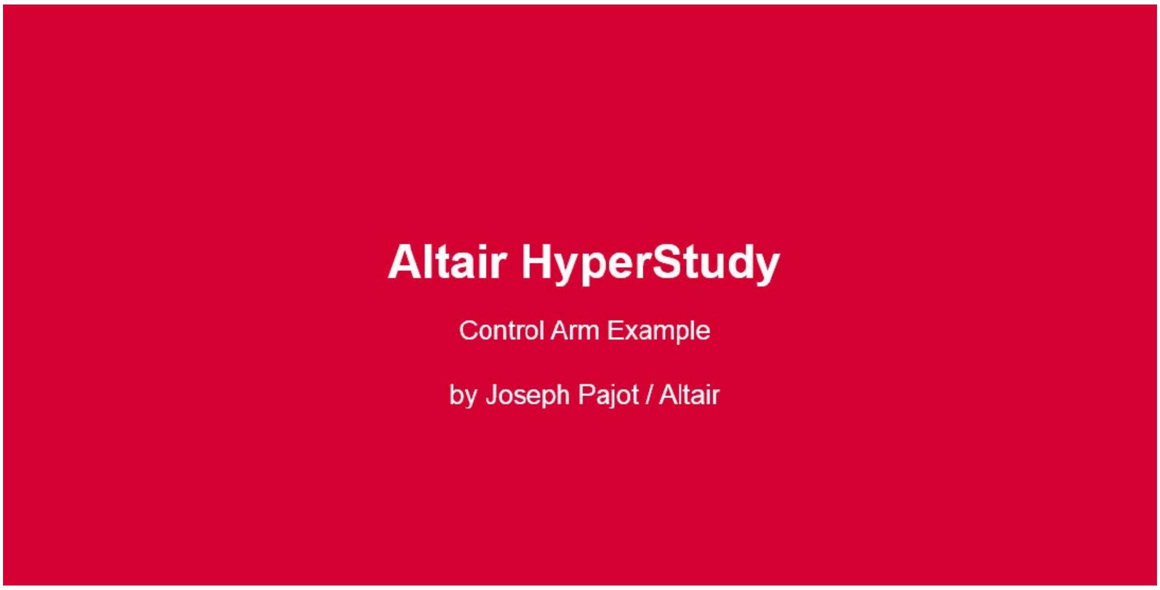
7.5.5 Comparison

In the table below, conclusions on the output response Max Stress is compared between the results obtained from the four DOE methods. As can be seen, the conclusion for Full Factorial and Fractional Factorial V are the same however fractional factorial resolution V required only 1/4th of the runs that full factorial did. You can also observe that as the resolution decreases, accuracy of the information gathered also decreases. For example, the conclusion from Fractional factorial resolution III is the least accurate.

Relation and Influence of DOE Methods on DV's	Full Factorial	Fractional Factorial V	Fractional Factorial IV	Fractional Factorial III	Plackett Burman	Fractional Factorial III 3 Level
Conclusions	Remove 3 radius	Remove 3 radius	Remove radius 1, 2 and height	Remove radius 2 and height	Remove radius 1 and 3	Remove radius 3 and height
Number of runs	512	128	24	12	12	27

Comparison of Results (Relation and Influence of DV's on Sum of Stresses) of Different DOE methods.

Note: In the video below (related to the Control Arm Model) Joseph Pajot takes you several steps further. He describes how DOE is used in the context of a reliability study (which in fact, represents a topic of its own and will be discussed in an upcoming version of this eBook)



https://altair-2.wistia.com/medias/o19kurm9fc

7.6 Bond Strength DOE

Earlier in this book we introduced the concept of DOE by referring to the Bond Strength Example/Project.

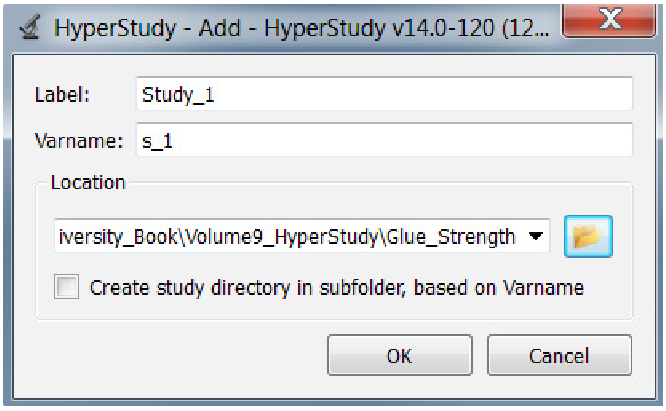
In the context of this example we demonstrate how to read the test data into HyperStudy.

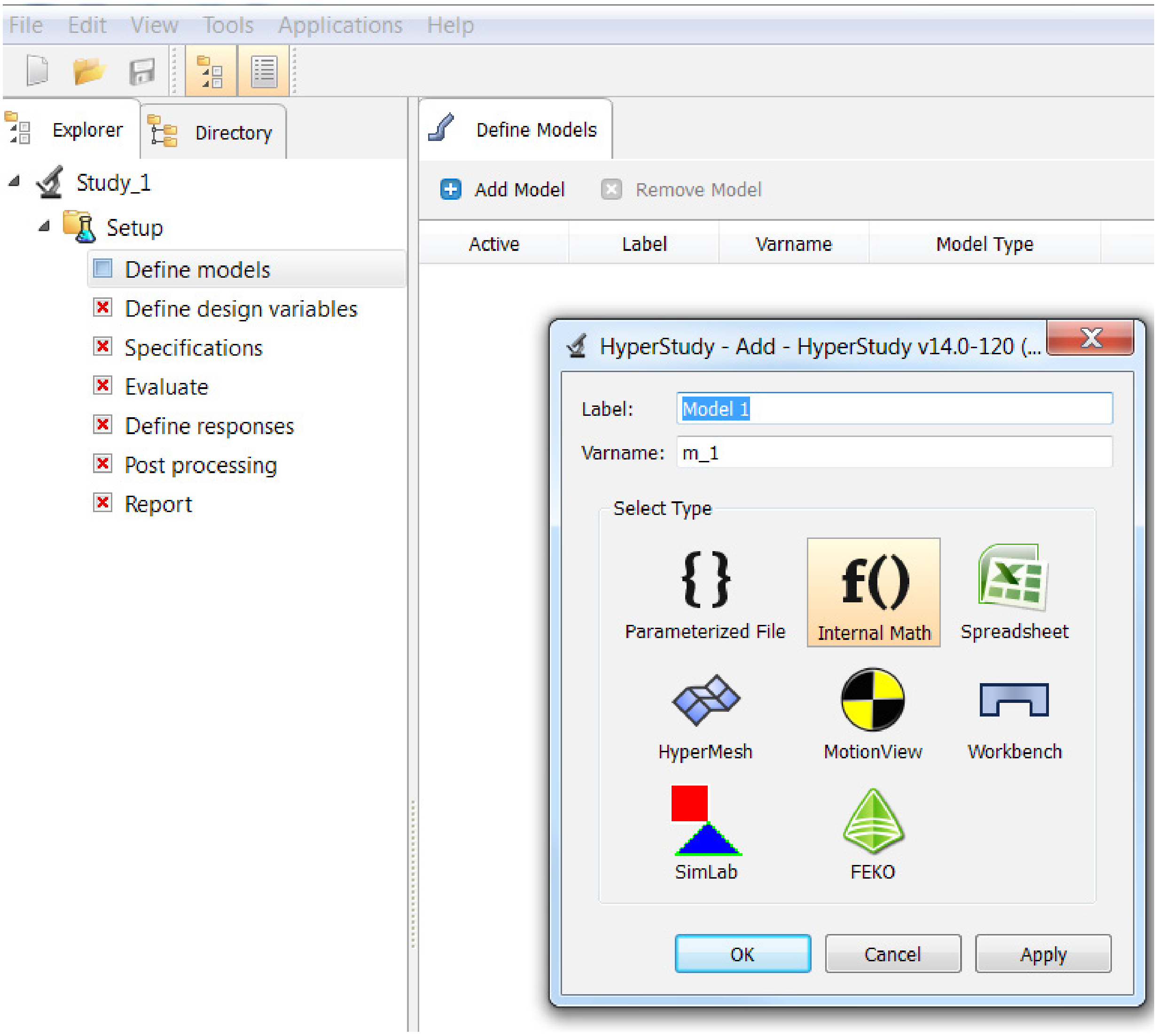
	Temperature	Pressure	Strength
Experiment# 1	100 degrees	50 psi	21 lbs
Experiment# 2	100 degrees	100 psi	42 lbs
Experiment# 3	200 degrees	50 psi	51 lbs
Experiment# 4	200 degrees	100 psi	57 lbs

Measured bond strength values

Like before we need to start with the Study setup.

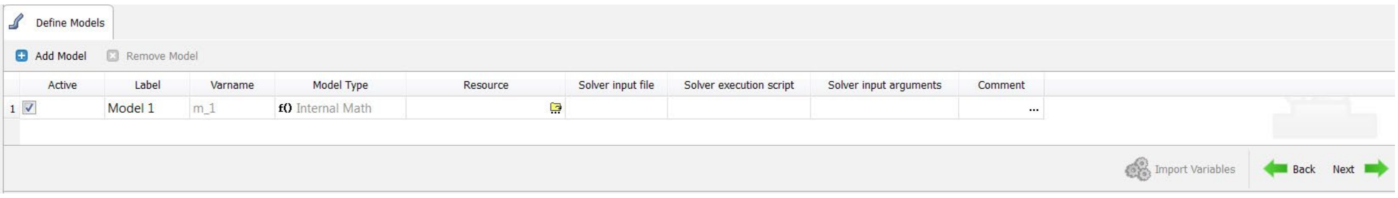
We begin with the definition of a new study



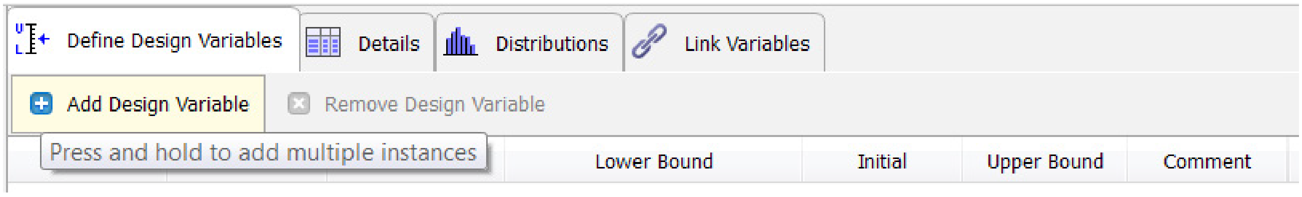


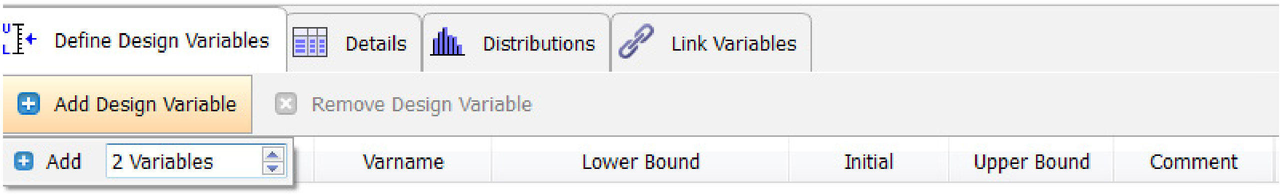
You may have compiled your test data in an Excel Sheet and hence would tend to select “Spreadsheet” as the Model Type.

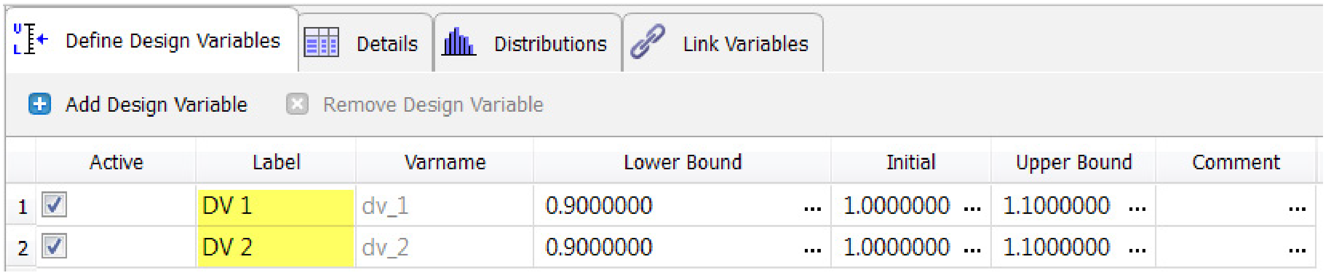
Please note, that “Spreadsheet” is used if some math happens inside the Excel sheet (e.g. multiplying fields). As we are dealing with a “static” data sheet (test data) we need to use “Internal Math. With this model type, you do not need to load a resource file or define the solver input file, solver execution script, or solver input arguments because you will create the design variables and responses in the “Define design variables” and “Define responses” steps.



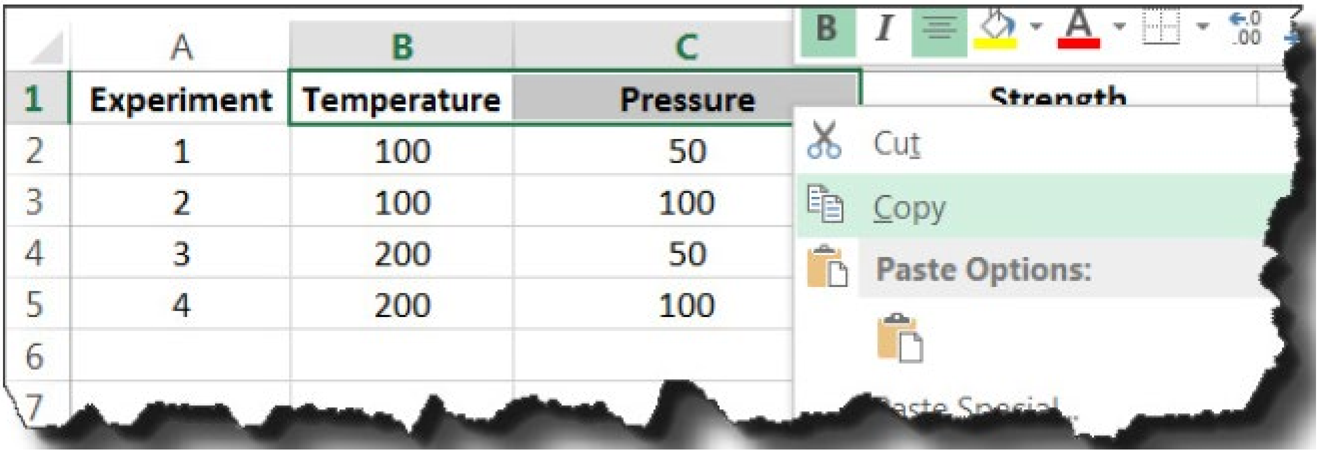
Next, we define the (two) Design Variables either by adding them one by one, or by “Press and hold” on the Add Design Variable tab.



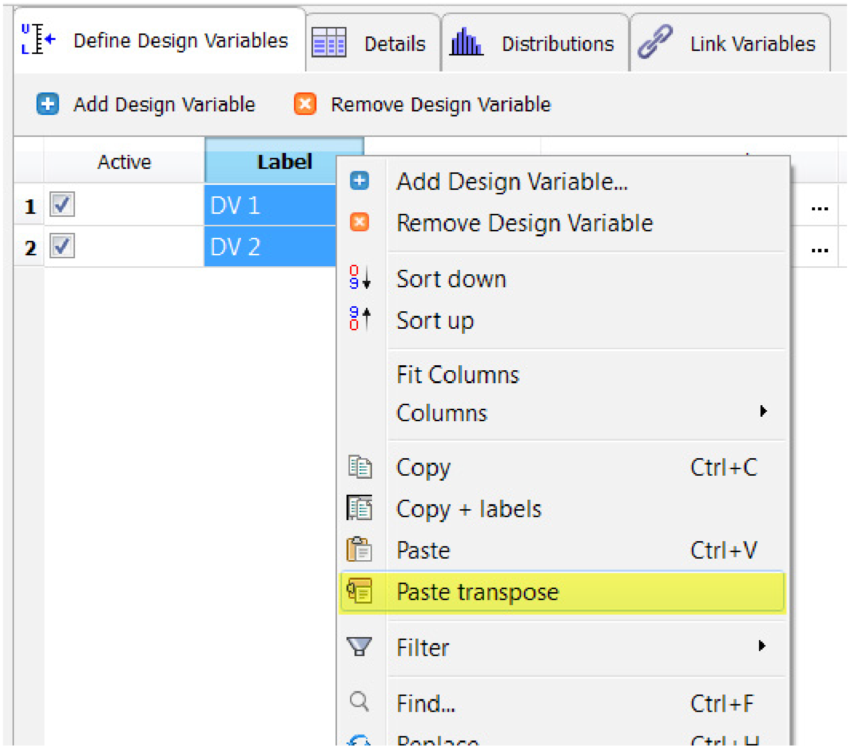




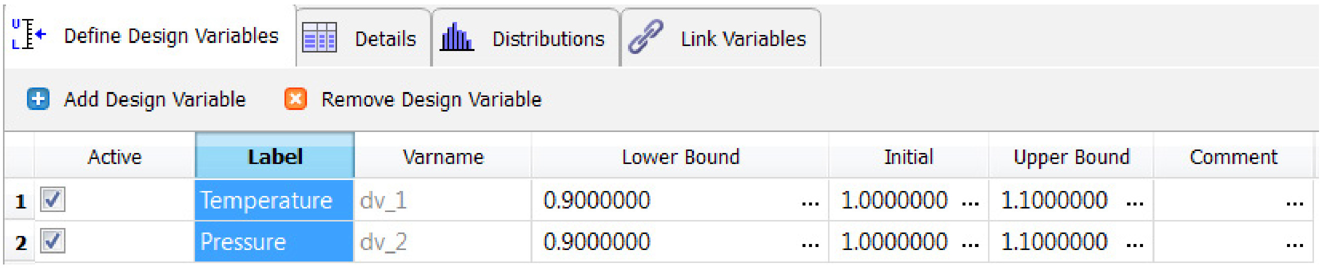
The two design variables are automatically named DV1 and DV2. Copy the header of the two design variables from the Excel Sheet (Temperature/Pressure),



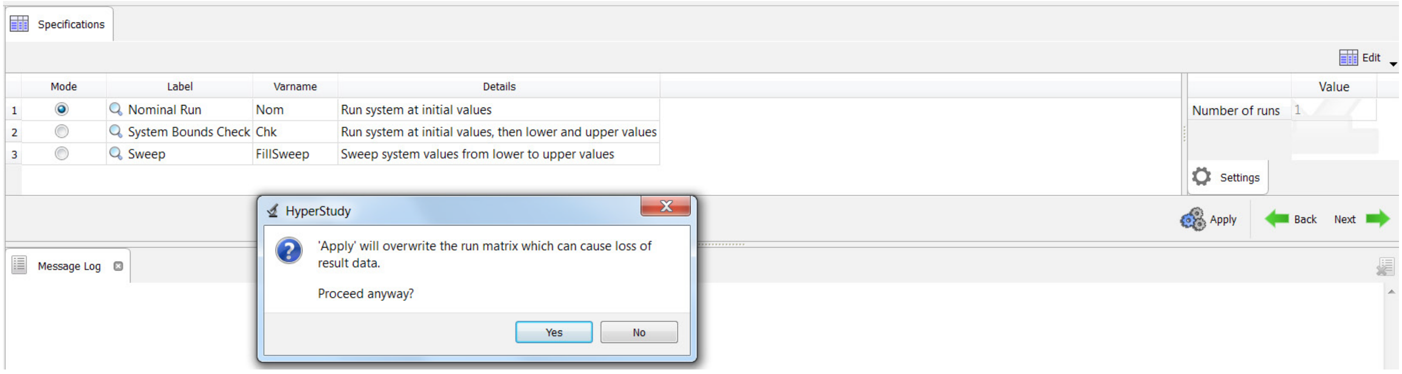
Then right mouse clicks on “Label” and select “Paste transpose”



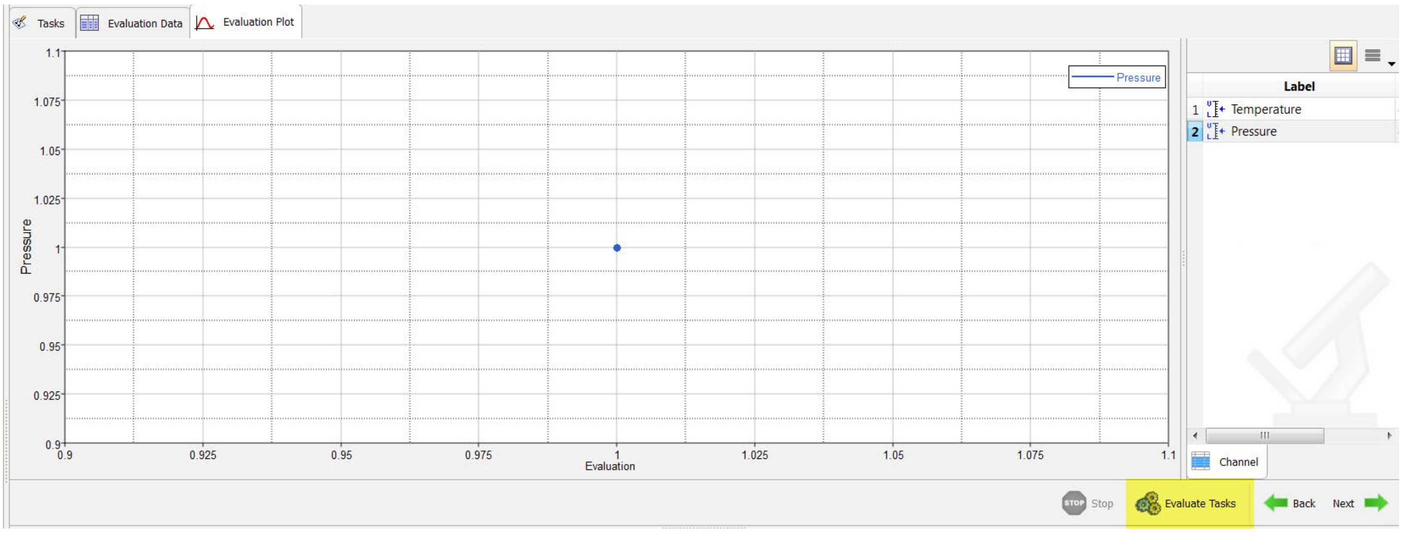
Which replaces the default names with the names used in the Excel Sheet.



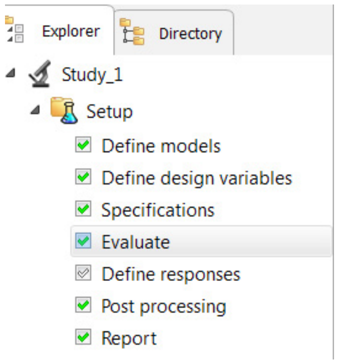
Next step is “Specification”. We select “Nominal Run” and confirm this strep with “Apply”.



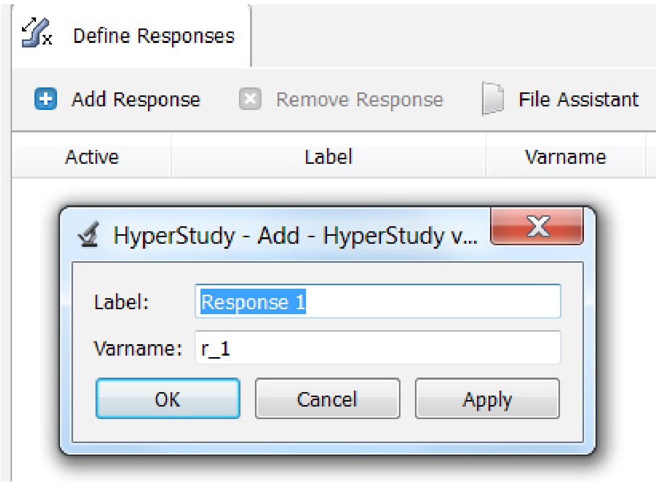
Don’t miss the next step: Evaluate Task (as the “Next” button is highlighted it may happen that you click on “Next” without pursuing the “Evaluate Step”).



In the “Explorer” Define responses, Postprocessing and Report can be accessed (if you miss the Evaluate Step from before you will not be able to advance to Postprocessing / Report)



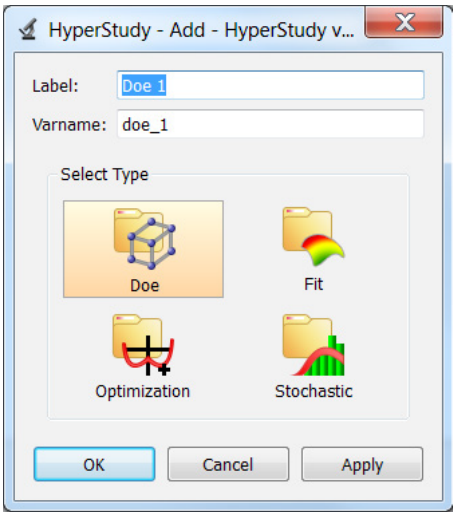
Next, we add the Response to the study.



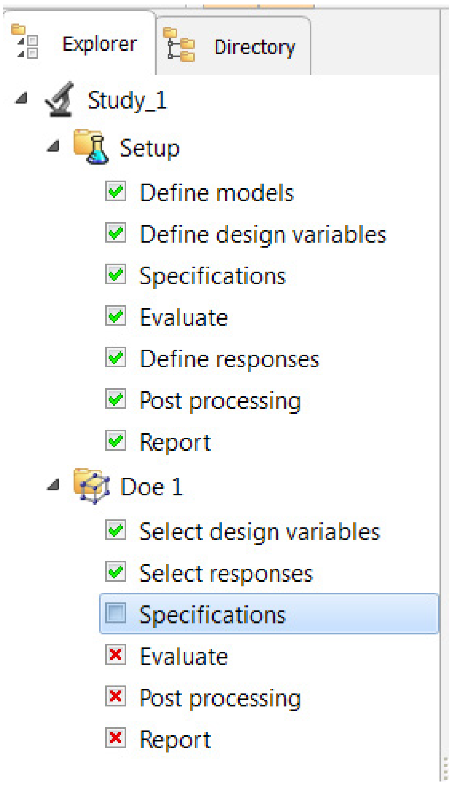
You may copy the response labels from the Excel spreadsheet, and paste them into the Label column in the work area. When you paste the design, variable labels into the work area, select “Paste transpose” from the context menu.

We now need to proceed to “Postprocessing”. The response data will be added in the DOE step which is next.

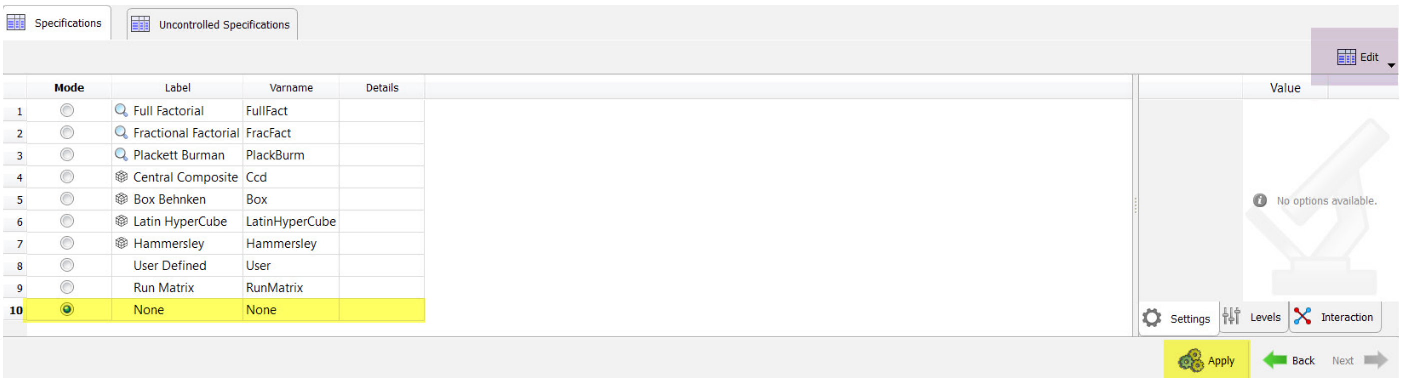
Add Approach: **DOE**



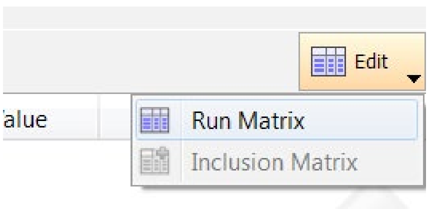
In the DOE step the first two tasks “Select design variables”, and “Select responses” are taken over from the Setup steps before.



We need to focus on “Specifications” next. Here we select “Mode” None. Don’t forget to click on “Apply”.



In the upper right corner of the Specifications tab, click on “Edit”

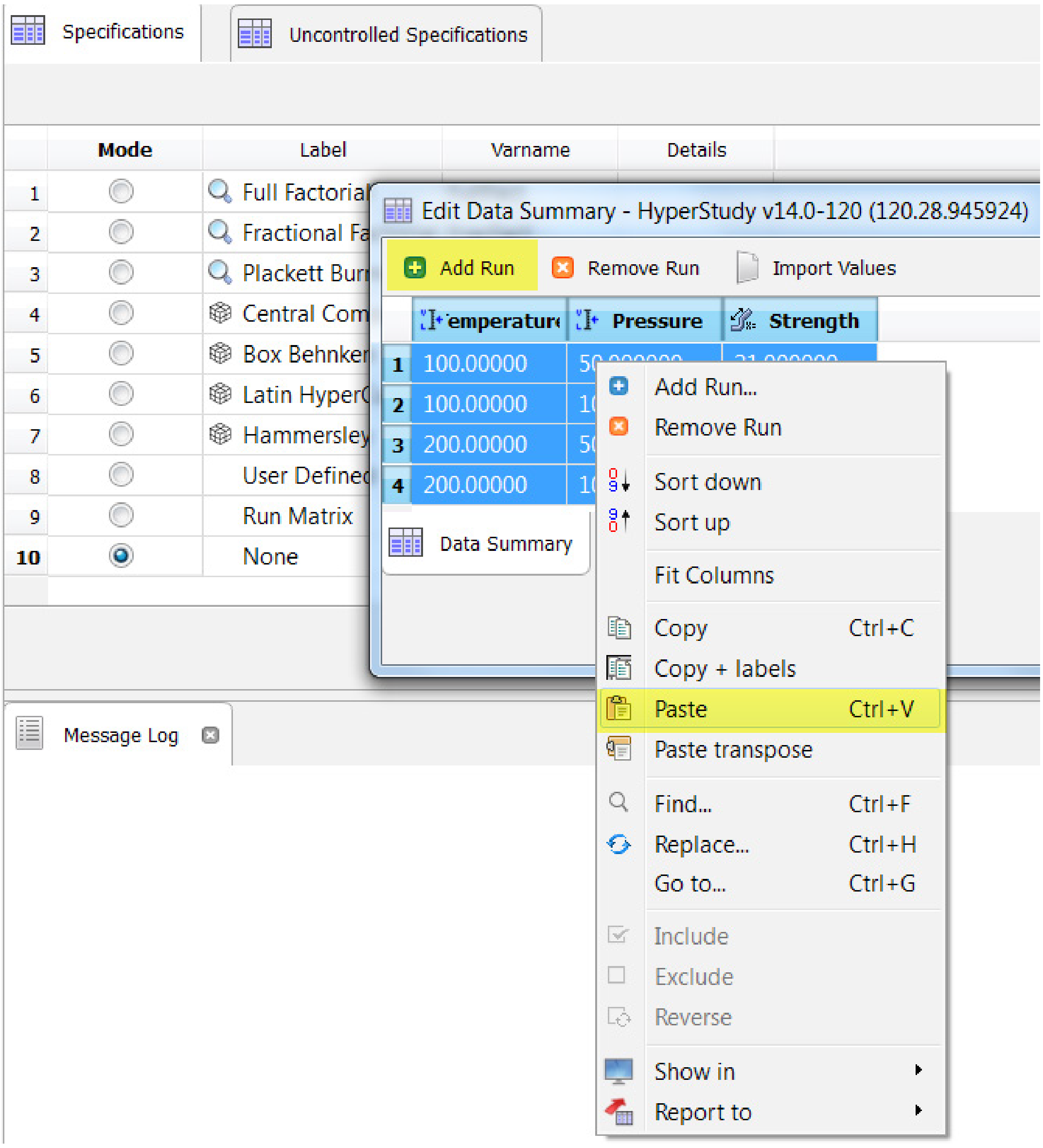


to edit the “Run Matrix”.

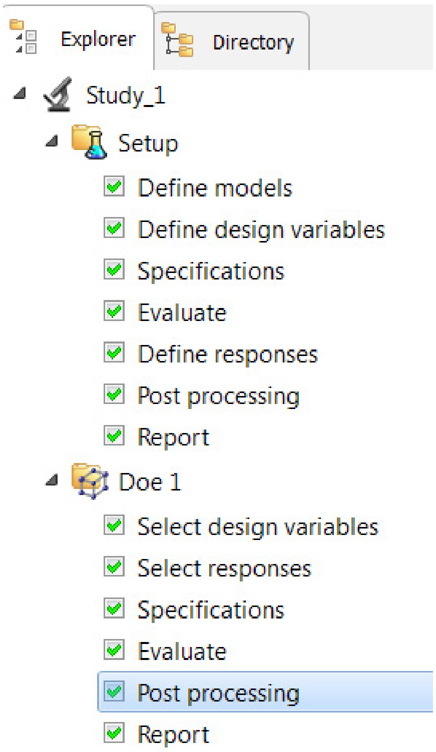
In the Run Matrix dialog, click Add Run to add 4 runs to the matrix, as there are 4 runs in the Excel spreadsheet. Copy all of the design variable and response data for each run in the spreadsheet.

	A	B	C	D
1	**Experiment**	**Temperature**	**Pressure**	**Strength**
2	1	100	50	21
3	2	100	100	42
4	3	200	50	51
5	4	200	100	57

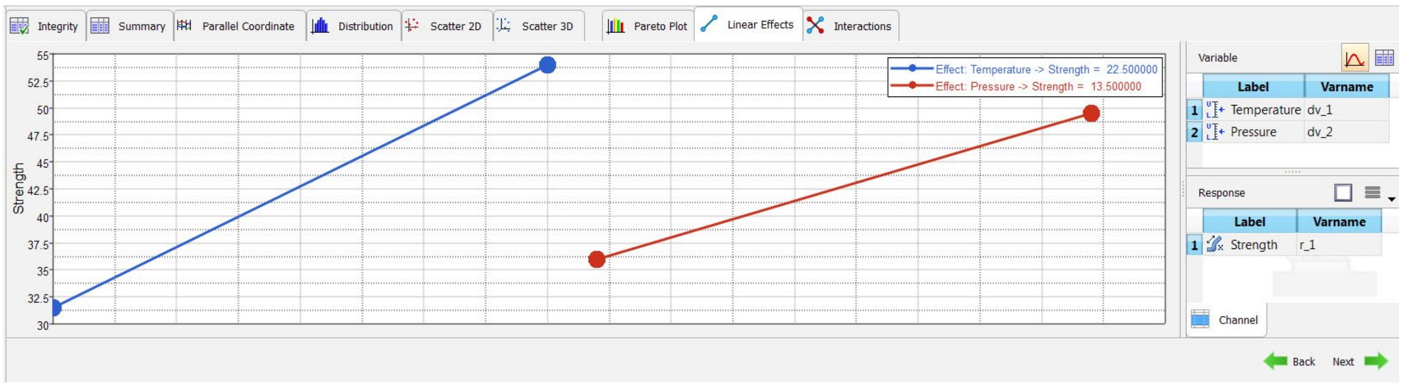
And “Paste” the data into the Run Matrix.



After the data were pasted into the Run Matrix, proceed straight to Postprocessing without clicking on Apply or Evaluate Task as all the data are already available in the Run Matrix.



In the Postprocessing step we look at the graphs shown earlier in this book, for instance, Main Effects plots of Temperature (blue) and Pressure (red) on Bond Strength.



7.7 HyperStudy for AcuSolve

The example below is available in the Online Help but we thought we should include at least parts of it in here in order to demonstrate the technical advantage of the following combination of tools:

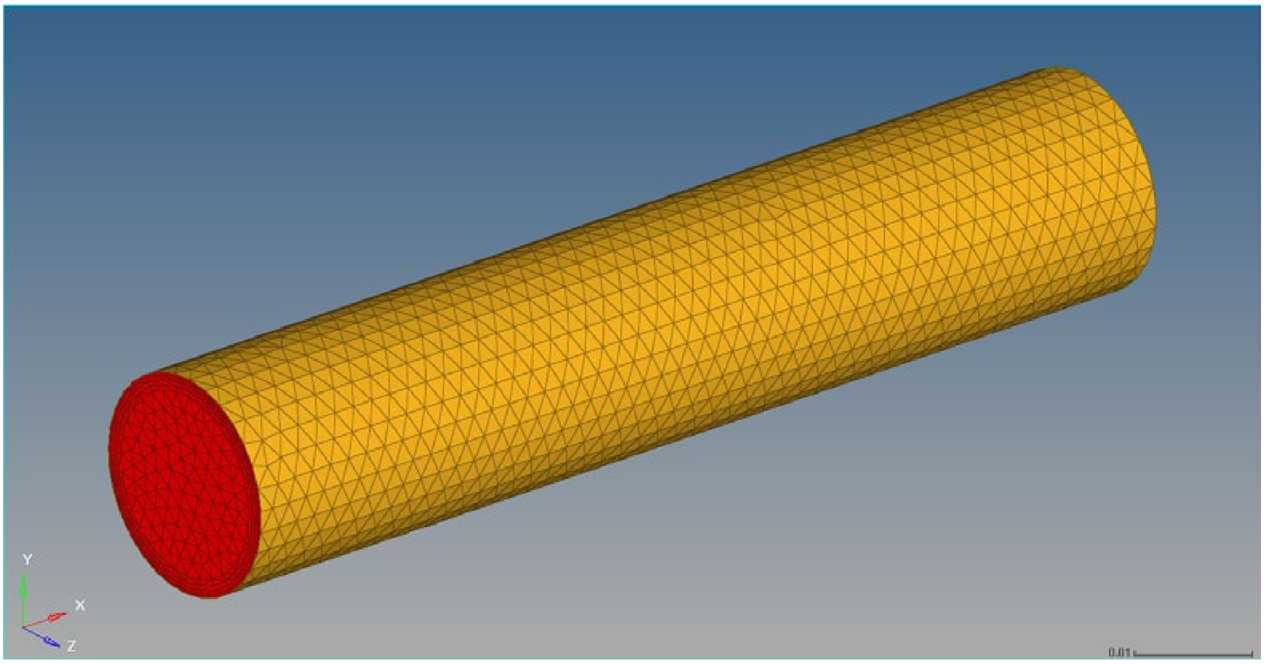
HyperStudy – Morphing (in HyperMesh) - CFD solver (here Acusolve).

The subject of the study is to analyze sensitivity of flow (i.e. the pressure drop between inlet and outlet) to changes in the shape (bending) of a pipe with our CFD solver AcuSolve.

Note: The process would be quite the same if you would use a non-altair CFD solver.

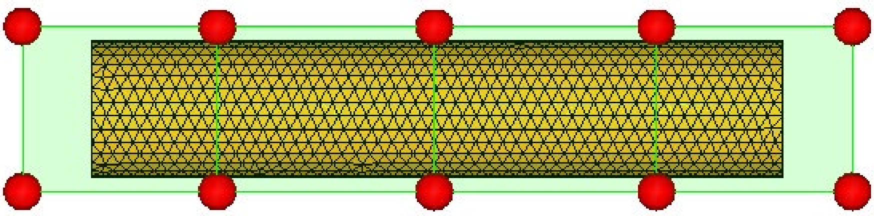
Step 1:

Parameterize the model using HyperMorph and HyperStudy

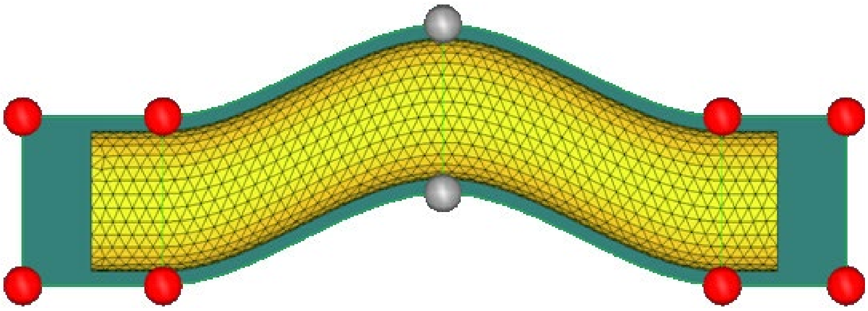


Base line model (pipe.hm)

The finite element model will be “wrapped” into a so-called morph-volume (which can be edited/splitted in smaller volumes easily).



The red dots/spheres refer to “Morph handles” which will be moved / translated in the next step.



Parameterized model (morphed model)

The shape change of the model (elements) is saved as a Shape Design Variable.

Step 2:

Use the HyperStudy Job Launcher to couple AcuSolve and HyperStudy

Step 3:

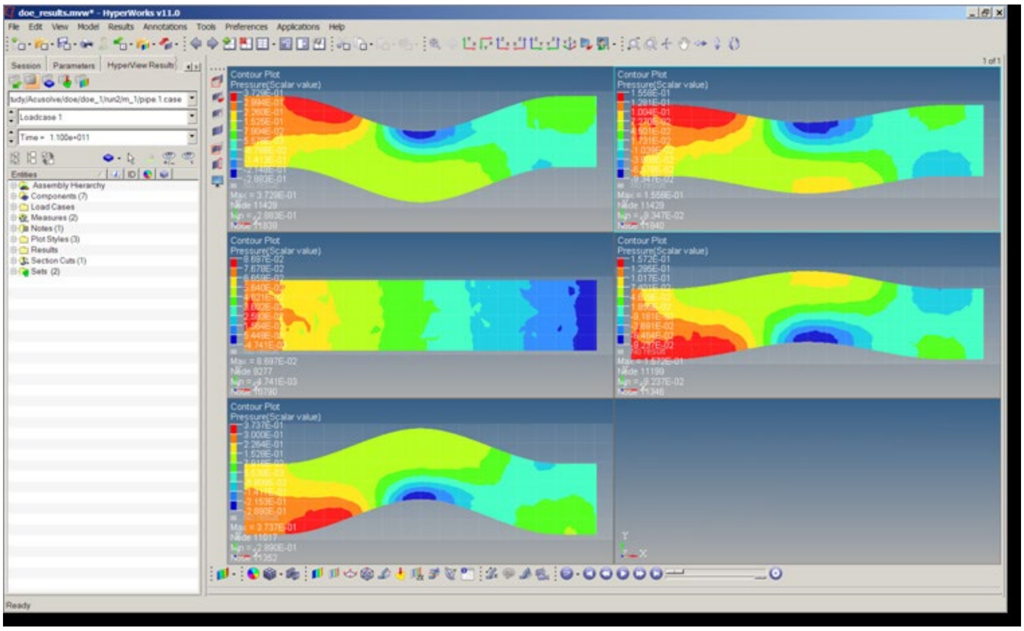
Set up and run a DOE study

The GUI of the HyperStudy Job Launcher allows to define, for instance, number of processors and the Responses.

After these decisions have been made a nominal run is submitted. Once the run is finished, HyperStudy opens and with the study Setup completed.

Do you recall the next steps after the study setup is complete?

We need to define the “Approach” next – which is a DOE (to study the sensitivity of flow (i.e. the pressure drop between inlet and outlet) to changes in the shape (bending) of the pipe).



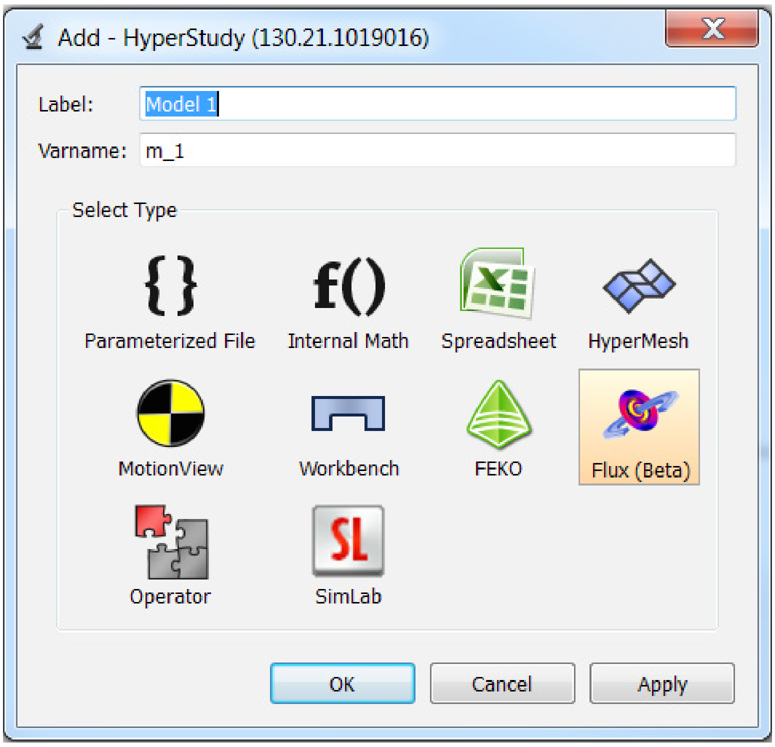
The example is contained in the Online Help:

HS-1535: Coupling HyperStudy, Morphing and AcuSolve (HyperStudy Job Launcher)

http://www.altairhyperworks.com/hwhelp/Altair/hw14.0/help/hst/hst.htm?hs_1535.htm

7.8 HyperStudy for Flux

The HyperStudy version 14.130 provides a new Flux model type. This model allows coupling HyperStudy and Flux Finite Element software and applying the HyperStudy approaches for Electromagnetic (EM) models design exploration and optimization.

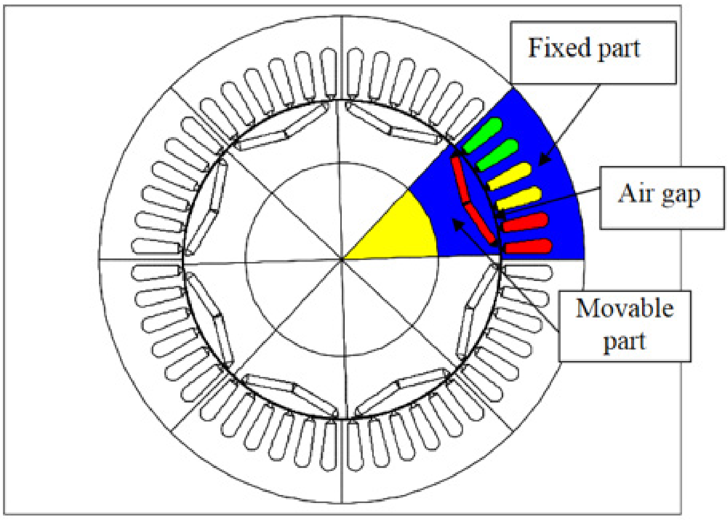


Define model window in HyperStudy

In this example one will show how to use HyperStudy to perform a DOE on a Flux model in order to screen the most influential parameters.

The studied device is a brushless AC embedded permanent magnet motor composed by:

- A fixed part (stator) including yoke, slots and windings.
- An air gaps.
- A movable part (rotor) with embedded magnets (in red on the picture)

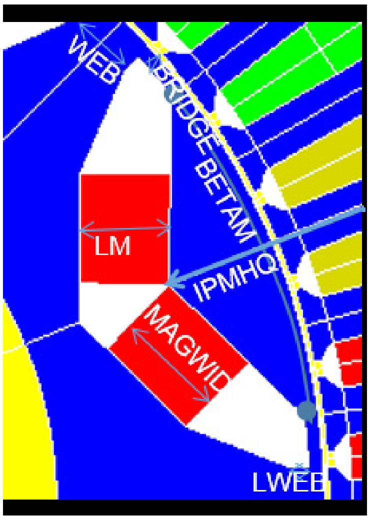


Motor geometry defined in Flux

This motor is designed for hybrid electric vehicle traction/generation with the following ratings:

- Max bus voltages: 500 V;
- Peak torque: 400 N.m;
- Max speed: 6000 rpm;
- Peak power ratings: 50 kW at 1200-1500 rpm.

The device has been previously modeled in Flux software. The geometry has been parameterized using 7 geometric parameters as shown on the picture below. Then it has been meshed and the physical properties have been defined. After that the model is solved to get the results that will be used as responses in HyperStudy.



Geometric parameters used

There are two responses of interest:

- Magnet surface (LM*MAGWID);
- Average torque (TORQUE_MEAN).

The first one does not really need a screening as it is defined by an analytical function depending of two parameters LM and MAGWID. For the second one we do not know which parameter has the biggest influence that is why one would like to perform a DOE to better understand the device operation.

	Definition	Units	Initial value	
BETAM	Magnet pole arc	mm	120	Design variables
LM	Magnet length	mm	12	
MAGWID	Magnet width	mm	30	
IPMHQ	Magnet cap depth	mm	15	
BRIDGE	IPM bridge	mm	2	
LWEB	Web radial length	mm	2,75	
WEB	IPM web	mm	10	
LM*MAGWID	**Magnet surface**	**mm²**	**360**	Responses
TORQUE_MEAN	**Average torque**	**N.m**	**360,12**	

Design variables and responses

Once the Flux model is ready one need to make two things in order to establish the connection with HyperStudy:

1. Create a link file from Flux.
2. Load this link file in HyperStudy.

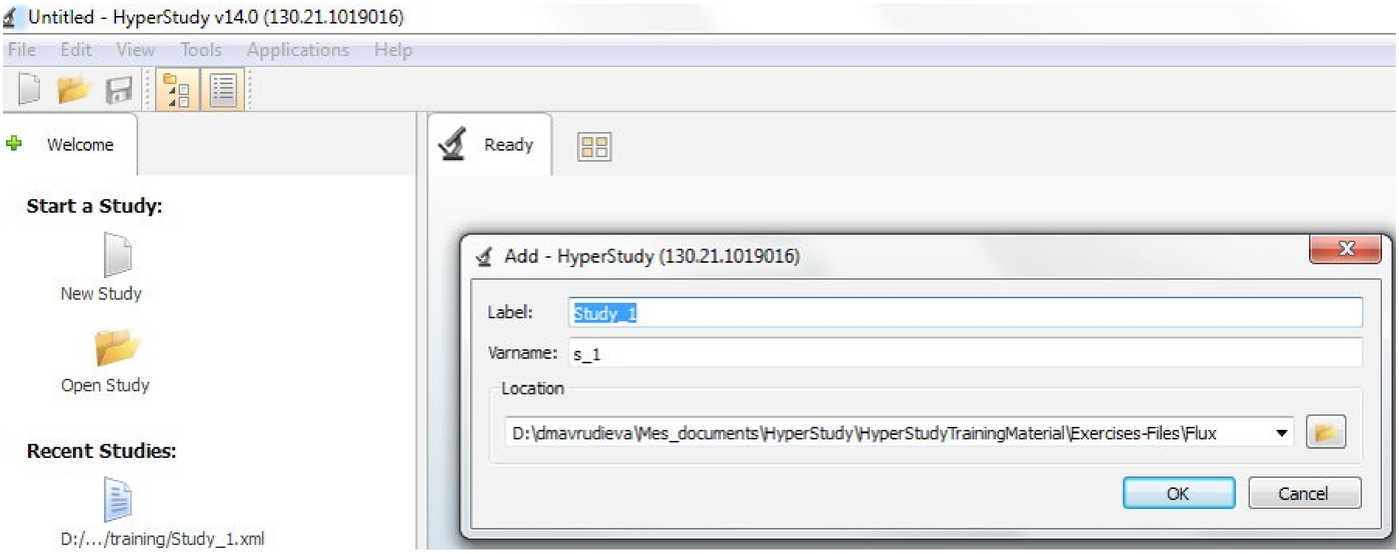
The first action is done in Flux using the command “Generate component for GOT-It coupling”* available in the menu Solving.

*Note: In Flux 2017 the command will be called “Generate component for HyperStudy coupling”.

This command allows selecting the input parameters (design variables) and the output functions (responses) from the Flux model.

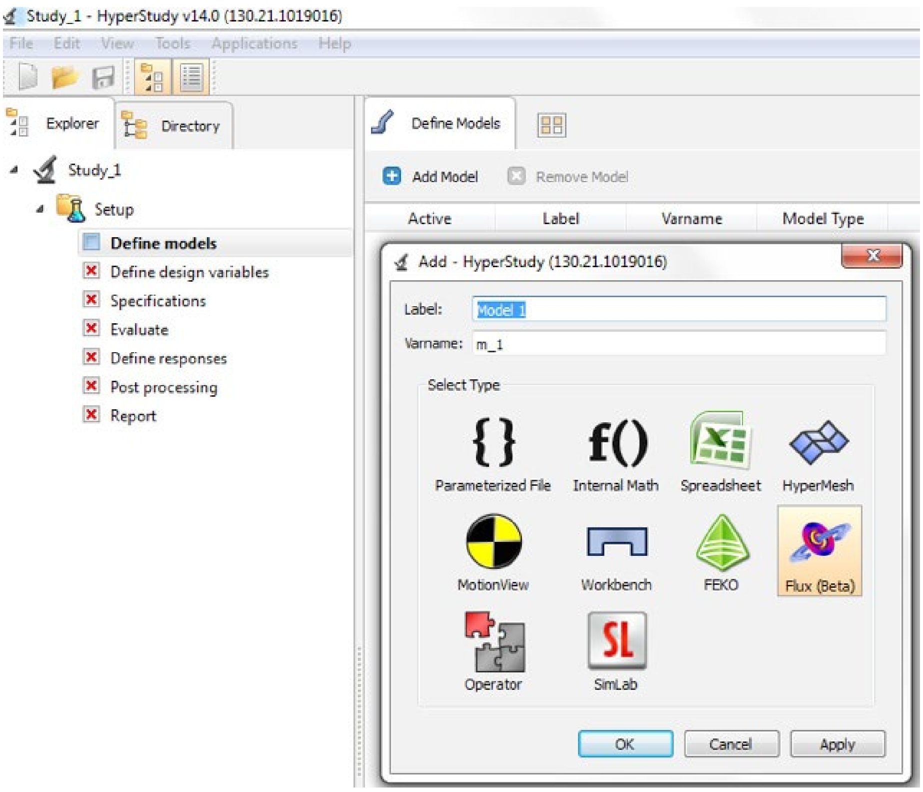
In order to send them to HyperStudy. As a result, it generates a link file (.F2G). Once the link file has been generated one could move on to HyperStudy in order to start the study.

Like in the other examples before, we start a new study (it is recommended to save the study in the same folder as the F2G link file).



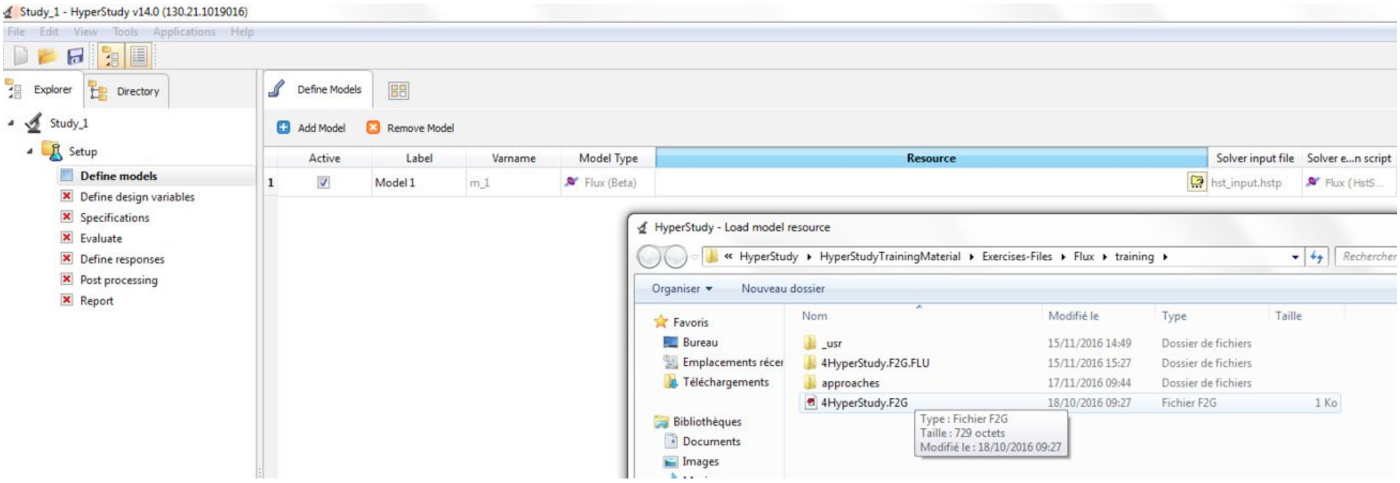
Start a new study

In order to create a Flux model, one need to click on Add Model and select Flux in the displayed window.



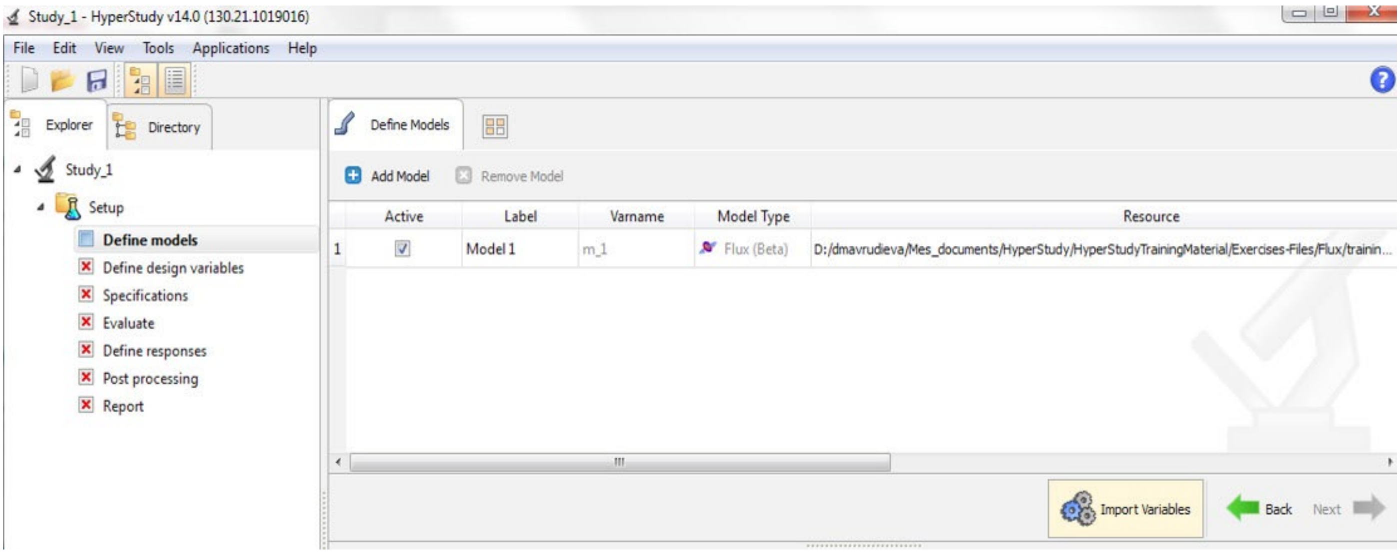
Model selection

Once the model has been added, click on Resource browser and select the F2G link file (you can also Drag & Drop it).



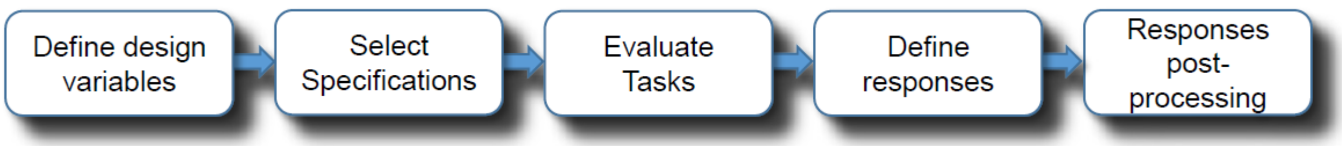
Flux model definition

Once the link file has been opened, one should click on Import Variables in order to import the design variables.



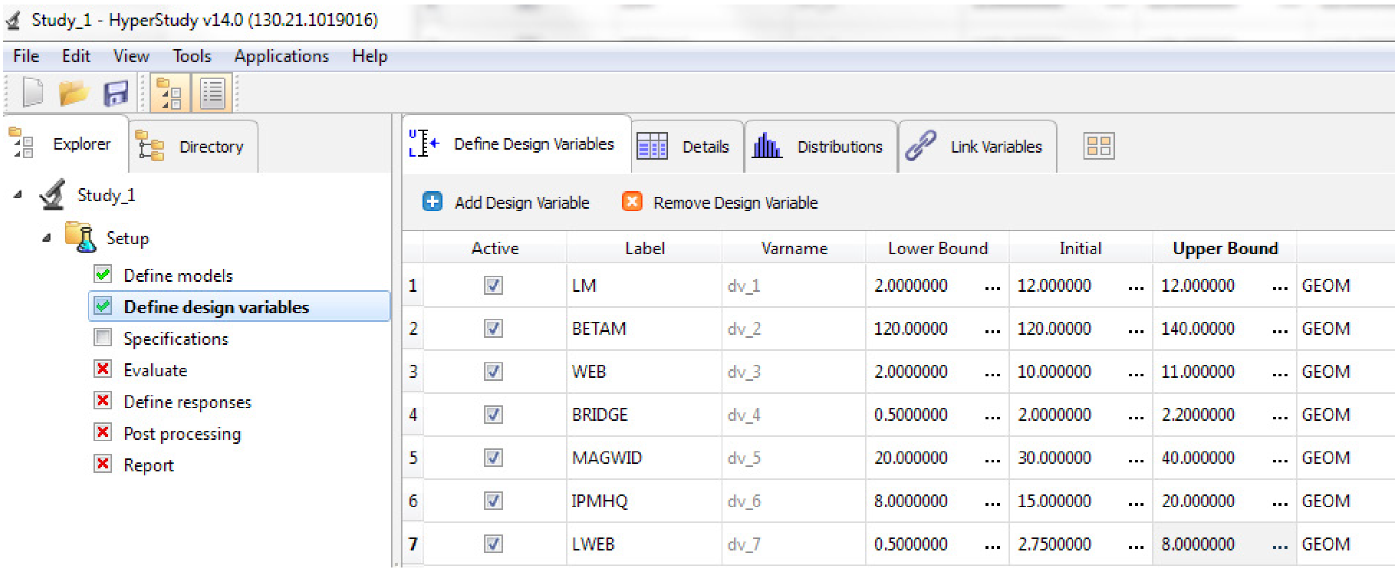
Design variables import in HyperStudy.

Now, the model is completely integrated in the study and one can continue defining the Setup by following the next steps:



Setup workflow

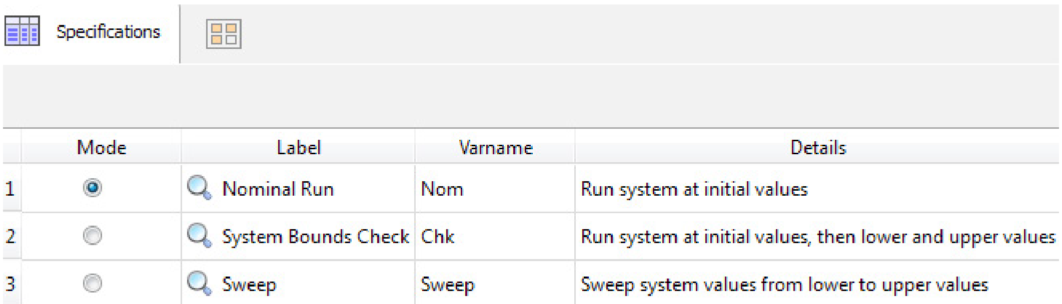
First, one should edit the lower and upper bounds of the design variables as shown on the picture below.



	Active	Label	Varname	Lower Bound	Initial	Upper Bound	
1	☑	LM	dv_1	2.0000000	12.000000	12.000000	GEOM
2	☑	BETAM	dv_2	120.00000	120.00000	140.00000	GEOM
3	☑	WEB	dv_3	2.0000000	10.000000	11.000000	GEOM
4	☑	BRIDGE	dv_4	0.5000000	2.0000000	2.2000000	GEOM
5	☑	MAGWID	dv_5	20.000000	30.000000	40.000000	GEOM
6	☑	IPMHQ	dv_6	8.0000000	15.000000	20.000000	GEOM
7	☑	LWEB	dv_7	0.5000000	2.7500000	8.0000000	GEOM

Design variables definition

After that step is taken, it is time to specify that we are interested in a Nominal Run. For that, move to the "Specifications" section and select Nominal run, then click on Apply.

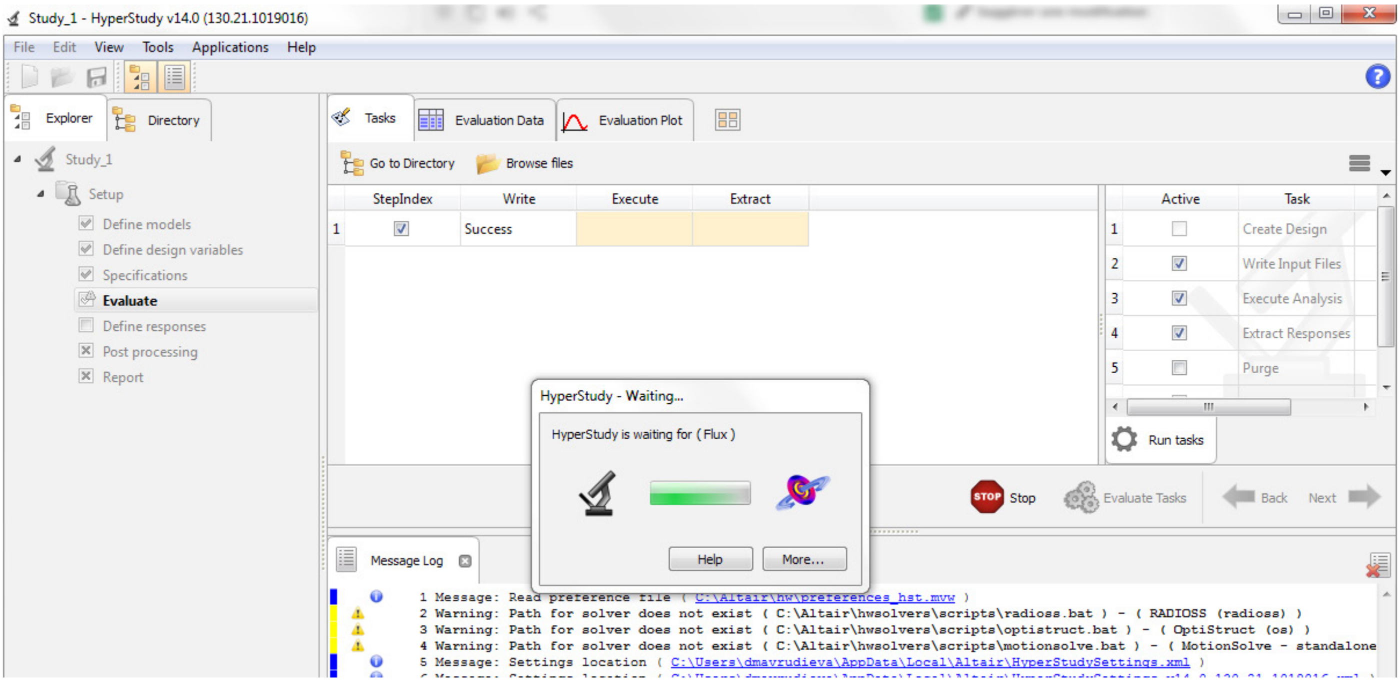


	Mode	Label	Varname	Details
1	◉	Nominal Run	Nom	Run system at initial values
2	○	System Bounds Check	Chk	Run system at initial values, then lower and upper values
3	○	Sweep	Sweep	Sweep system values from lower to upper values

Specifications definition

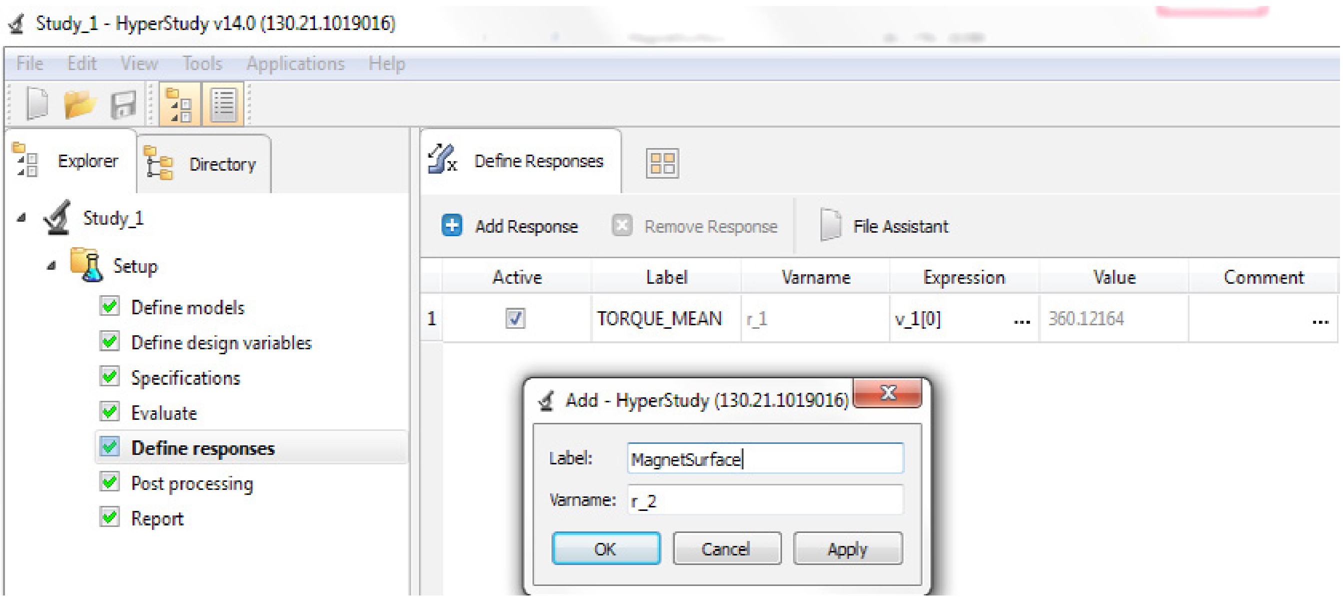
You can click on Next (in the right down corner) in order to switch between different sections.

In Evaluate section, click on "Evaluate Tasks" in order to perform the nominal run. You will see such kind of window telling you that HyperStudy is launching Flux in batch mode to make the simulation and extracts responses.



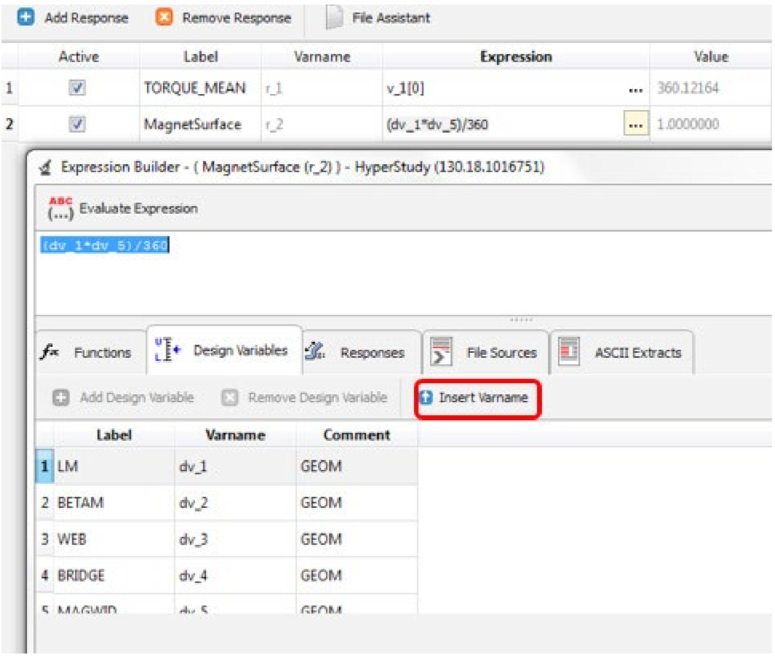
Flux launching from HyperStudy

Once the run is finished you can see in "Define responses" section that TORQUE_MEAN response (performance coming from Flux) has been automatically created. One will create a second response for the magnet surface by using the Expression builder. To start, click on Add response button.



Add a new response magnet surface

Once the new response has been added click on the "…" in the Expression tab in order to open the Expression builder. In the Design Variables tab, select LM and click on Insert Varname, then insert MAGWID as well in order to define the function (LM*MAGWID)/360.



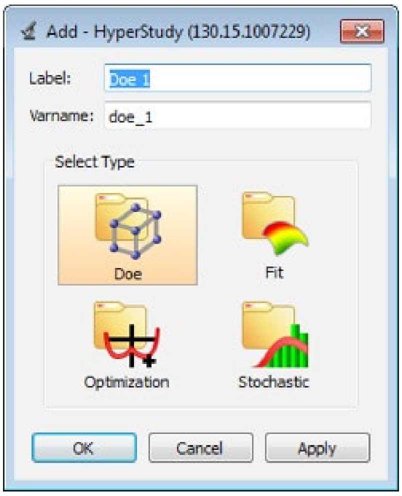
Magnet surface function definition

Note: Compare "Label" and "Varname" (listed in the image above) with the expression (dv_1*dv_5)/360 which corresponds to "MagnetSurface = (LM*MAGWID)/360"

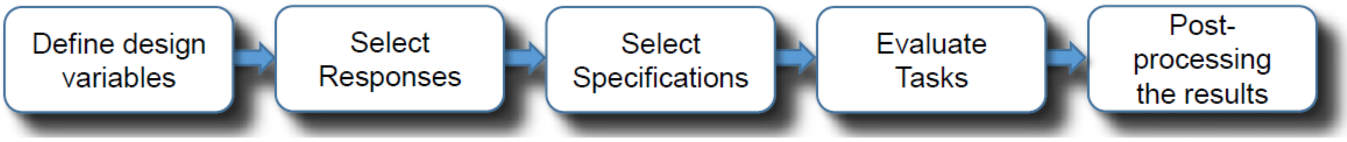
As stated in the other examples, do not forget to click on "Insert Varname" after each DV selection.

The setup definition is now done and one could add a new approach. In the next stage, we will define a DOE approach in order to perform a screening on the responses.

We start by adding a new approach DOE and then continue following the screening workflow.



Add a new approach DOE



Screening workflow

The design variables definition is kept same as for the Nominal Run (see reminder below).

Add Design Variable   Remove Design Variable

	Active	Label	Varname	Lower Bound	Initial	Upper Bound	Comment
1	☑	LM	dv_1	2.0000000 ...	12.000000 ...	12.000000 ...	GEOM ...
2	☑	BETAM	dv_2	120.00000 ...	120.00000 ...	140.00000 ...	GEOM ...
3	☑	WEB	dv_3	2.0000000 ...	10.000000 ...	11.000000 ...	GEOM ...
4	☑	BRIDGE	dv_4	0.5000000 ...	2.0000000 ...	2.0000000 ...	GEOM ...
5	☑	MAGWID	dv_5	20.000000 ...	30.000000 ...	40.000000 ...	GEOM ...
6	☑	IPMHQ	dv_6	8.0000000 ...	15.000000 ...	20.000000 ...	GEOM ...
7	☑	LWEB	dv_7	0.5000000 ...	2.7500000 ...	8.0000000 ...	GEOM ...

Design Variables definition for DOE screening

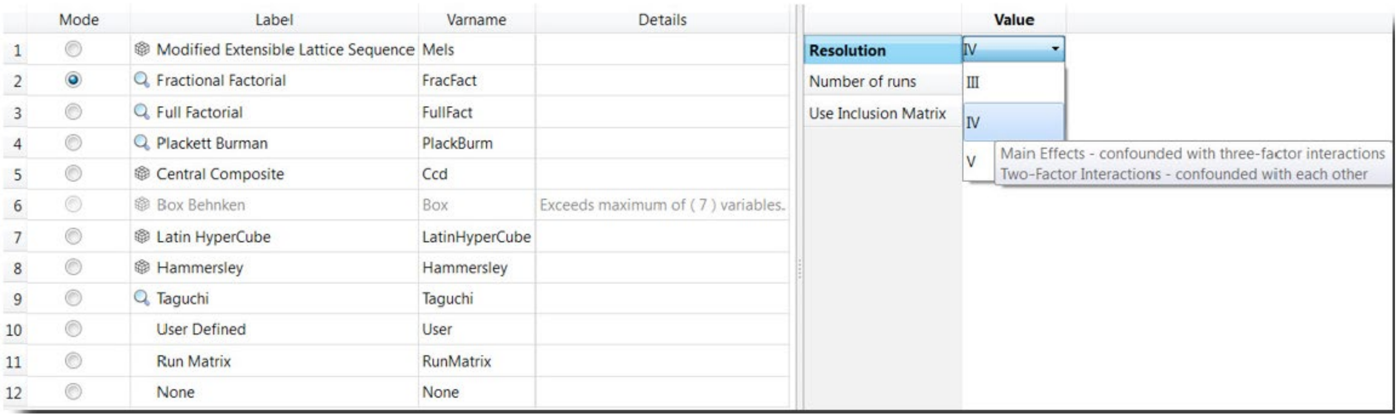
The main response of interest in TORQUE_MEAN, but we select both responses.

	Active	Label	Varname	Expression	Evaluate From
1	☑	TORQUE_MEAN	r_1	v_1[0]	SOLVER
2	☑	MagnetSurface	r_2	(dv_2*dv_3)/360	SOLVER

Responses selection for screening

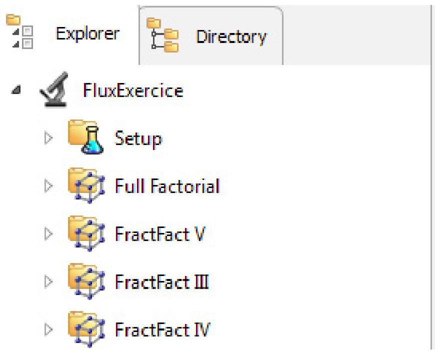
We perform the following DOE in order to compare the results and the number of runs:

- Full Factorial
- Fractional Factorial Resolution III
- Fractional Factorial Resolution IV
- Fractional Factorial Resolution V



Fractional Factorial definition

After the DOE definition has been done, click on “Apply” and then “Evaluate tasks”. Note that each method needs to be started separately. It means that you have to add new approach DOE for each DOE used. You can name the different approaches in order to recognize them in the Study Explorer.

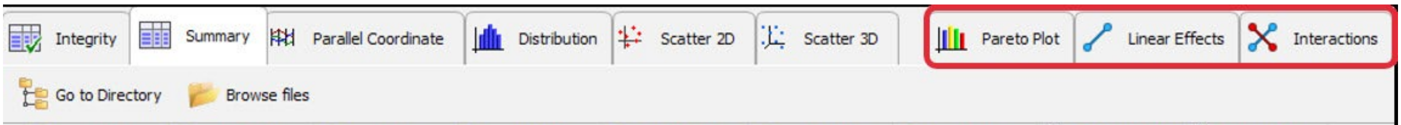


DOE creation in the data tree

Compare this process with the one described in the “DOE Method Comparison: Arm Model”.

In order to post process, the results we will consider the following displays:

- Linear effects.
- Pareto plots.
- Interactions.

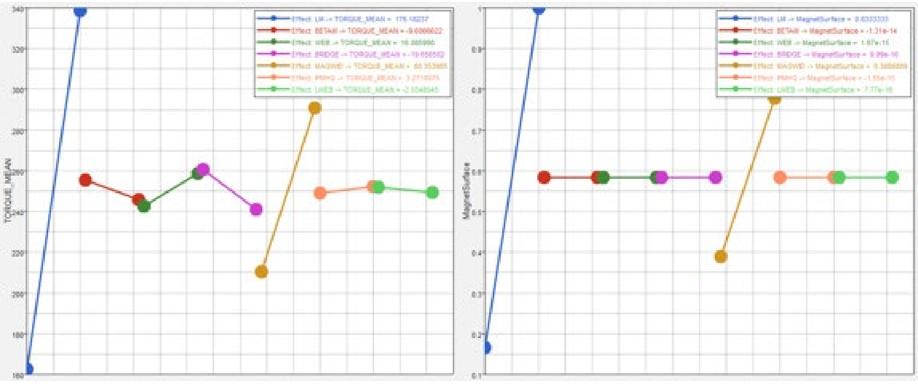


	LM	BETAM	WEB	BRIDGE	MAGWID	IPMHQ	LWEB	ORQUE_MEAI	MagnetSurface
1	2.0000000	120.00000	2.0000000	2.0000000	40.000000	20.000000	0.5000000	174.95731	0.2222222
2	2.0000000	120.00000	11.000000	2.0000000	20.000000	8.0000000	8.0000000	144.27905	0.1111111
3	2.0000000	140.00000	2.0000000	0.5000000	40.000000	8.0000000	8.0000000	174.97477	0.2222222
4	2.0000000	140.00000	11.000000	0.5000000	20.000000	20.000000	0.5000000	132.28534	0.1111111
5	12.000000	120.00000	2.0000000	0.5000000	20.000000	20.000000	8.0000000	300.95654	0.6666667
6	12.000000	120.00000	11.000000	0.5000000	40.000000	8.0000000	0.5000000	446.08294	1.3333333
7	12.000000	140.00000	2.0000000	2.0000000	20.000000	8.0000000	0.5000000	254.69266	0.6666667
8	12.000000	140.00000	11.000000	2.0000000	40.000000	20.000000	8.0000000	367.11465	1.3333333

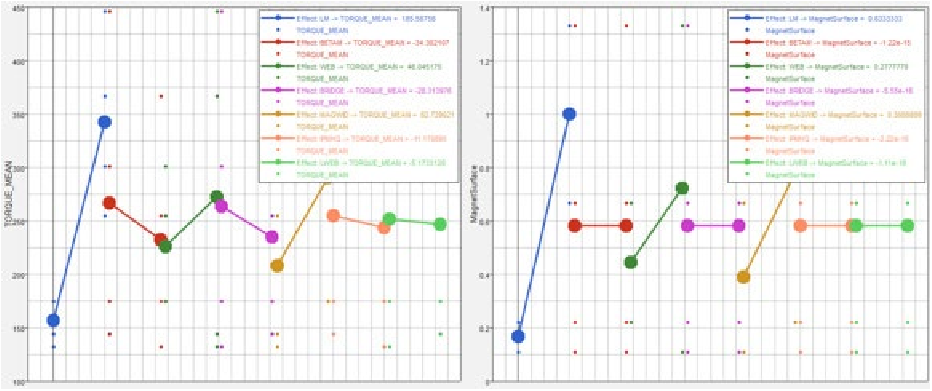
Postprocessing tabs

First, we compare the Linear effects (effects of design variables on TORQUE_MEAN and Magnet Surface) provided from Full Factorial and Fractional Factorial (III, IV or V). On the picture below one can see that LM and MAGWID have the biggest influence on both TORQUE_MEAN and Magnet Surface functions. Other influential parameters on TORQUE_MEAN are BRIDGE and WEB.

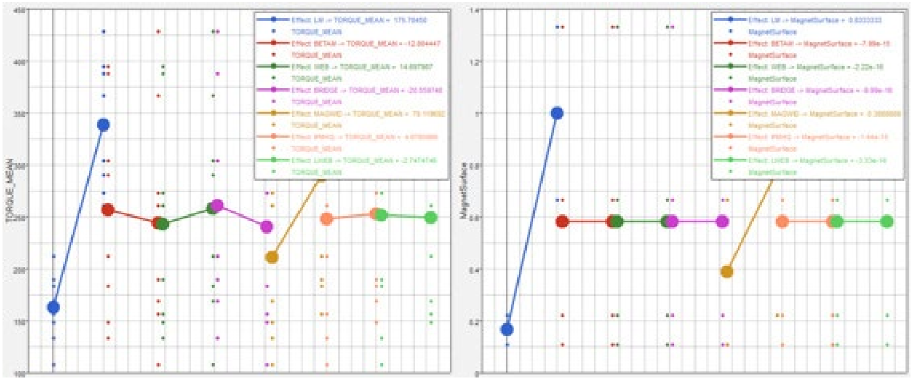
The most accurate information is provided from Full Factorial as it runs a complete number of sets (128 in this case). But one would like to know if we can get the same information with less runs using Fractional Factorial and what is the most appropriate Resolution type for Fractional Factorial.



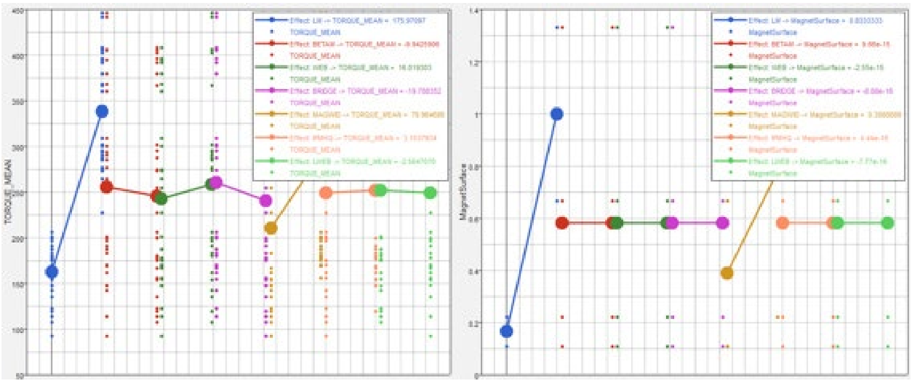
Linear effects provided from Full Factorial



Linear Effects obtained with Fractional Factorial III



Linear Effects obtained with Fractional Factorial IV

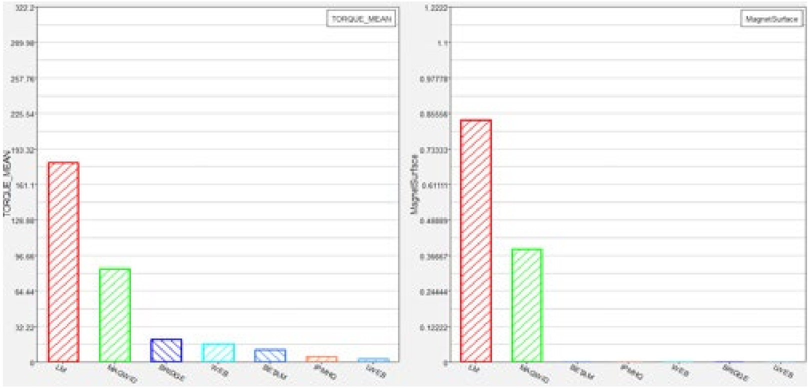


Linear Effects obtained with Fractional Factorial V

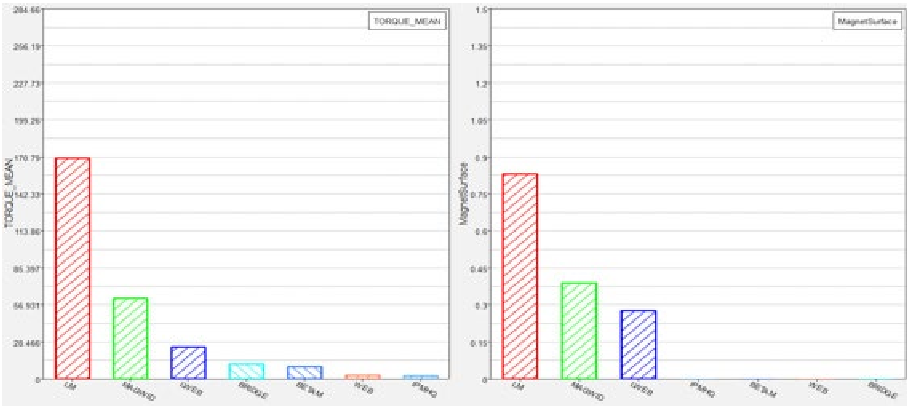
We can notice that the results obtained with Fractional Factorial IV and V are similar to those obtained with Full Factorial, with much less runs (16 runs for Fractional Factorial IV and 64 runs for Fractional Factorial V). On the other hand when using Fractional Factorial III the parameter effects on TORQUE_MEAN are increased. Also, there is significant effect of WEB parameter on Magnet Surface. This effect should not be as Magnet Surface depends only on LM and MAGWID. In that case this effect is due to a confounding.

So, looking at the “Linear effects” Fractional Factorial IV seems to be the most efficient choice.

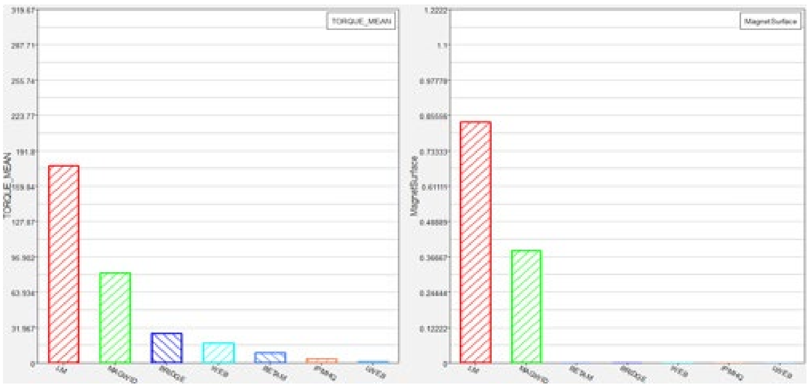
Let’s make now the same comparison using the Pareto plots.



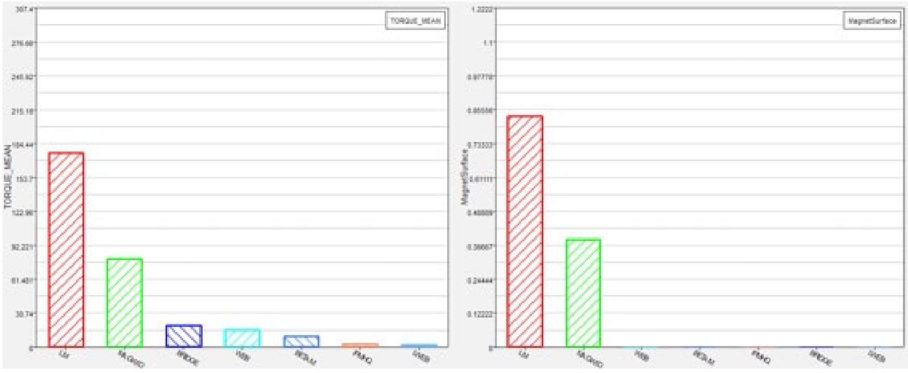
Full Factorial Pareto plots



Fractional Factorial III Pareto plots



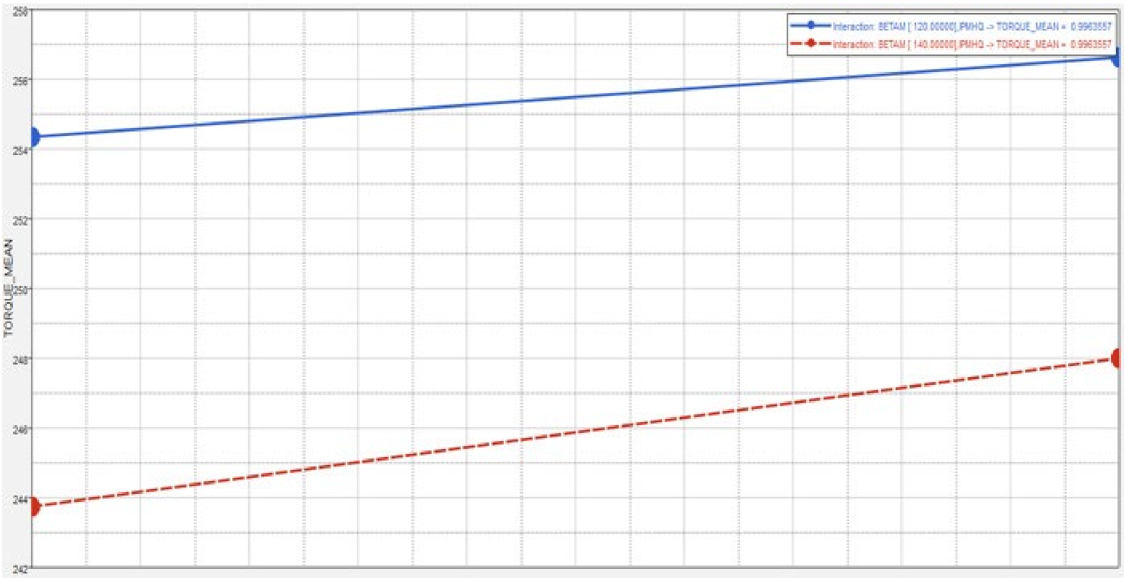
Fractional Factorial IV Pareto plots



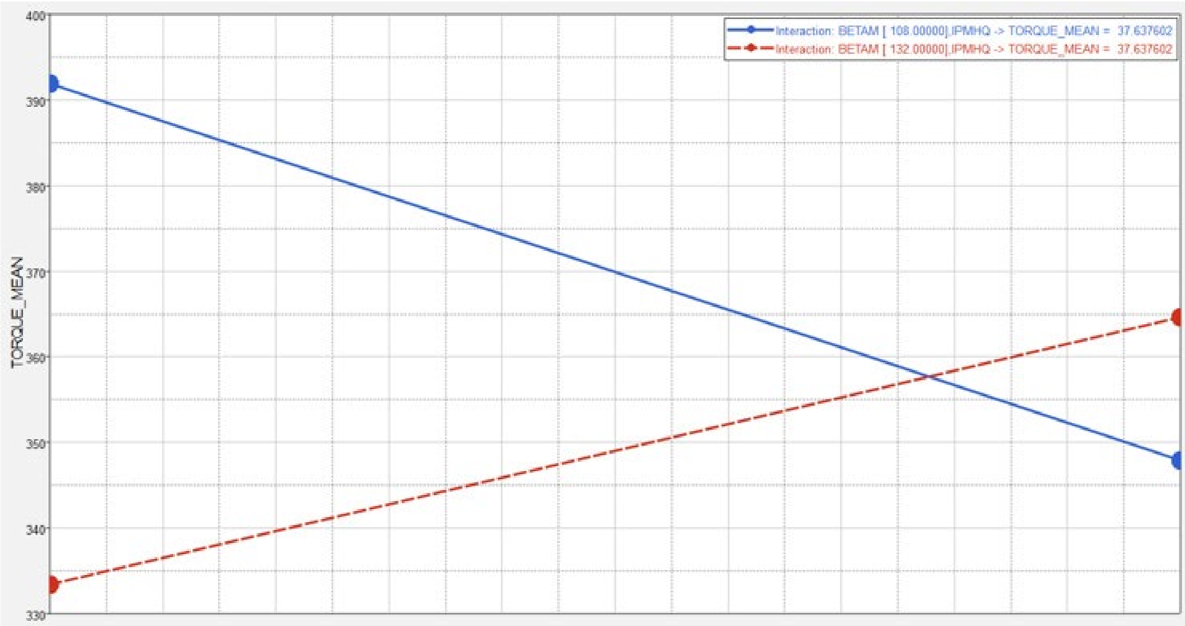
Fractional Factorial V Pareto plots

Same as for the “Linear effects”, Fractional Factorial IV and V provides similar results as Full Factorial, but not Fractional Factorial III.

What about the “Interactions” comparison? Let’s compare the same interaction with Full Factorial and Fractional Factorial III.



Interaction between BETAM and IPMHQ on TORQUE_MEAN obtained with Full Factorial

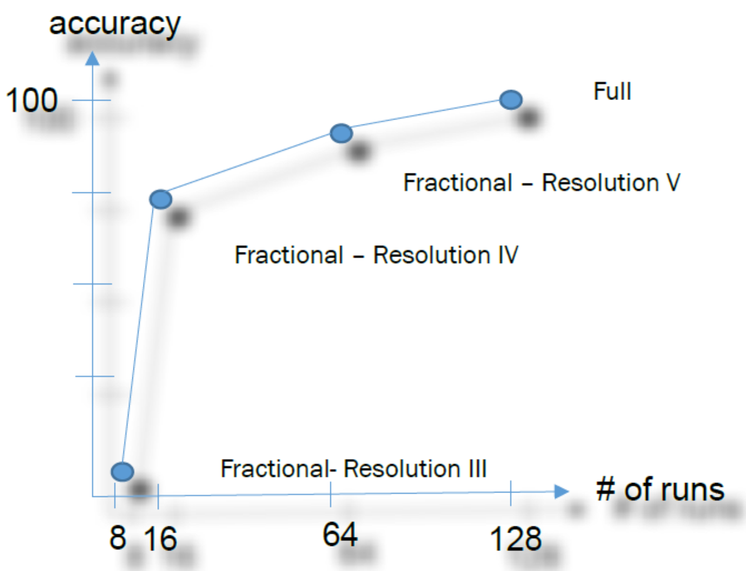


Interaction between BETAM and IPMHQ on TORQUE_MEAN obtained with resolution III Fractional Factorial

As one can see, an amplitude interaction has been detected with Fractional Factorial III but not with Full Factorial. The right information is that provided by Full Factorial. Once again, there is confounding effects when using Fractional Factorial III.

In this example we have seen that DOE can be easily run on Flux models. The use of different DOE methods has been compared using Linear effects, Pareto plots and Interactions charts. To help the user to choose the right DOE method, let's consider the graph below that illustrates the accuracy vs effort plot for all the DOE used. The most accurate but also the most expensive is Full Factorial (128 runs). Fractional Factorial III (8 runs) is the cheapest one but it leads to some inaccuracy and confounding. Fractional Factorial IV (16 runs) and V (64 runs) have provided the same results as Full Factorial.

So, in terms of efficiency Fractional Factorial IV seems to be the best choice.



Accuracy vs effort

8 Appendix

8.1 Glossary

Approach (Study Approach)

Astudy approach is a certain set of steps taken to study the mathematical model of a design. In HyperStudy, there are four approaches: DOE, Fit, Optimization and Stochastic. Each approach serves a different design study purpose and the required steps for each approach may be unique. For example, you can use the DOE approach if you need to learn the main factors affecting your design, but you need to use the optimization approach if you want to find the design that achieves the design objectives while

satisfying design requirements.

Approximation

Expression that relates the response of interest back to the design variables that were varied. HyperStudy provides polynomial least square regressions, Moving Least Squares Method, and HyperKriging to define approximations. An approximation is only as good as the levels (number of designs) 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.

Box-Behnken (used for Space Filling)

- Used when functions are known to be second order
- Only 3 levels
- No interactions
- Puts designs in the centers, extremes are ignored
- For 3 DV's; 13 designs (FuF has 27)

Design of Experiments (DOE)

Is a structured, organized design/run matrix creation method that once run will be used to determine the relationships between the different factors/variables (Xs) affecting a process and the output of that process (Y).

Often used to represent a parameter study using a Design of Experiments. The DOE is actually the way the design variable combinations for the study are chosen. It is a method to perform a parameter study with a minimum number of designs (experiments) to obtain as complete an information as possible about the design space. The opposite would be a full-factorial analysis (complete design matrix), with all possible design variable combinations selected. A parameter study is used to study the behavior of design in the given design space.

HyperStudy provides the following Designs of Experiments (individual methods are explained in some detail in this eBook):

- Full factorial
- Fractional factorial
- Plackett-Burman

- Box-Behnken
- Central Composite Design
- Hammersley
- Latin Hypercube

Central Composite (used for Space Filling)

- Used when functions are known to be second order
- Only 5 levels
- 5 different types, inscribed or not, number of center points
- All interactions

Confounding

Confounding occurs when two factors are associated with each other or "travel together" and the effect of one is confused with the effect of the other. For example, in order to improve team performance, a soccer coach asks his team to run two miles a day while the players decide to take vitamins. In this case the effects of running two miles a day and taking vitamins will be confounded since it will not be possible to identify the effect of them on team performance independently.

In other words, Confounding is the inability to distinguish between main effects and interaction effects.

Confounding occurs when a fractional design is chosen instead of a full-factorial design.

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.

Design Variable / Factors

System parameter that influences the system performance in the chosen response.

A design variable is a property of the study. It is an object that is varied by the study based on certain rules. You can associate a design variable to a particular model parameter. Two design variables cannot be associated with the same model parameter. A design variable does not always need to be associated to a model parameter. Design variables can be linked within or across models.

Or in other words:

Design Variables are system parameters that change the system performance.

Design input (variable) whose value is changed to cause a change in the output (response).

Factors can be controlled or uncontrolled. They can be controlled or uncontrolled. A factor can be either Discrete i.e., slow (-) or fast (+) or Continuous i.e. temperature.

Effects

Are the changes in the responses produced by changes in the levels of the design variables. If the effects line is horizontal, it implies that the design variable has no effect on the response of interest. As the line becomes more vertical, the effect on the response of interest is greater. A positive slope indicates that changing the parameter value will result in a direct change to the response, whereas a negative slope indicates an inverse association.

Filling / Fitting (Approximation)

Fitting functions (Approximations) are meta models that represent the actual responses. Some simulations are computationally expensive which makes it impractical to rely on them exclusively for design studies. In these cases, use of approximations leads to substantial savings of computational resources.

Full Factorial (used for Screening)

- Used for parameter screening
- Design Variables with any levels
- All Interactions
- No confounding

Fractional Factorial (used for Screening)

- Used for parameter screening when cost of full factorial is prohibitive
- Can be 2 or 3 levels
- Interactions can be chosen

Hammersley (used for Space Filling)

- Used for metamodeling
- Number of designs is the number of levels

Interactions

Are the varying effects that a design variable can have on a response at varying levels of other design variables. An interaction is the failure of one factor to produce the same effect on the response at different levels of another factor. In other words, the strength or the sign (direction) of an effect is different depending on the value (level) of some other variable(s). An interaction can be either positive or negative. Graphically, two factor interactions are indicated by a significant non-parallelism of the two lines in an interaction plot.

Latin Hypercube (used for Space Filling)

- Used for metamodeling
- Number of designs is the number of levels

Levels

Levels are discrete (or continuous) values of the factor/variable. The values taken by x in the range [-1; +1] are called levels. So if you look at the example discussed earlier, the minimum and maximum temperature would correspond to -1 and +1. The number of levels per variable to be considered depends on the level of non linearity in the problem; for a linear model two levels are sufficient; for a quadratic model three levels are needed.

With respect to the example from above, if we would have considered not only 100 degrees and 200 degrees but also 150 degrees the study would have been a 3 level DOE.

Linear Effects

Shows the effect of a change in the level of a design variable on the response. Linear effects are shown in table and plot format. In a plot, if the line is horizontal, it implies that the design variable has no effect on the response of interest. As the line becomes more vertical, the effect is larger. A positive slope indicates that changing the parameter value will result in a direct change to the response. A negative slope indicates an inverse association.

Main Effect

Is the influence of individual factors on responses. Main effect of a factor is the change in response produced by a change in the level of the factor alone, averaged across the levels of other factors. Variables with small effects can be screened out without much loss of accuracy in further design studies such as optimization. This reduces the problem dimension which in turn will reduce the computational expenses as well as the amount of data to be investigated.

Optimization

Design optimization is the use of a collection of mathematical procedures to achieve a design as fully perfect, functional, or effective as possible. Quite often the optimized design represents a compromise between partially conflicting objectives.

Plackett Burman (used for Screening)

- Used when only main effects are important
- Only 2 levels
- No interactions
- For 3DV's 12 designs (FuF has 8); for 4DV's 12 designs, can also be 12, 20, 24, 28, 36 designs

Reduced Variables

In DOE terminology, it is standard practice to work with reduced variables that has a range of -1 to 1 for each real variable.

Reduced variables are associated to real variables with the following formula:

$$x = \frac{X - \frac{X_{MAX} + X_{MIN}}{2}}{\frac{X_{MAX} - X_{MIN}}{2}}$$

where X is the initial variable and x the reduced variable.

Regression

Is a polynomial (expression) that relates the response of interest back to the parameters 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 will result in more runs.

A Linear Regression model:

$$F(X) = a_0 + a_1X_1 + a_2X_2 + (error)$$

An Interaction Regression Model:

$$F(X) = a_0 + a_1X_1 + a_2X_2 + a_3X_1X_2 + (error)$$

A Quadratic Regression Model (2nd order):

$$F(X) = a_0 + a_1X_1 + a_2X_2 + a_3X_1X_2 + a_4X_1^2 + a_5X_2^2 + (error)$$

Response

Measurement of system performance. Examples are: weight, volume, displacement, stress, strain, reaction forces, and frequency

Response Surface

In the absence of a continuous function relating the objective to design variables, numerical experiments can be used to generate a table of objective-function values vs. design-variable values. A surface fitted through this table of points is called the Response Surface.

Robust Design

Understand whether the model (and method) is robust to small variations in the factor levels. If the model turns out to be non-robust then understand the bounds may be modified to achieve robustness.

A robust design is insensitive to small design changes. It minimizes the variations in performance and puts design on a predefined target.

There are two types of robust design:

- Variation in noise factors.
- Variation in design variables

Note: HyperStudy has the capability for performing robust design by combining an optimization study with a stochastic study.

Screening

Used at the beginning of the experimental phase. The objectives are twofold: 1. Explore/test as many factors as possible in order to understand whether they have an influence on the design. 2. Understand their appropriate ranges.

Sensitivity

Rate of change. Normally gradient of the response with respect to the design variable(s).

Study

In HyperStudy, a study is a self-contained project (saved into a study file, with a .xml extension) in which models, variables, responses, and approaches are defined. The HyperStudy GUI was designed to guide you through a series of steps to set up a study and/or to add an approach. Each step must be completed before progressing on to the next step.

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, humidity and/or occupant seating positioning. Of course, if the uncontrolled factors have an impact on the results they need to be considered in the study as well.

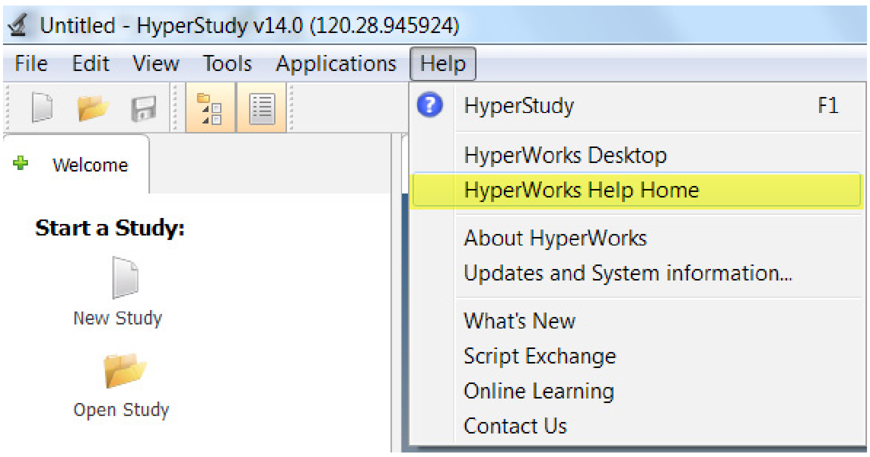
Variable

Also called factor. Anything that we can control or measure in an experiment. In engineering design we usually use the term design variable.

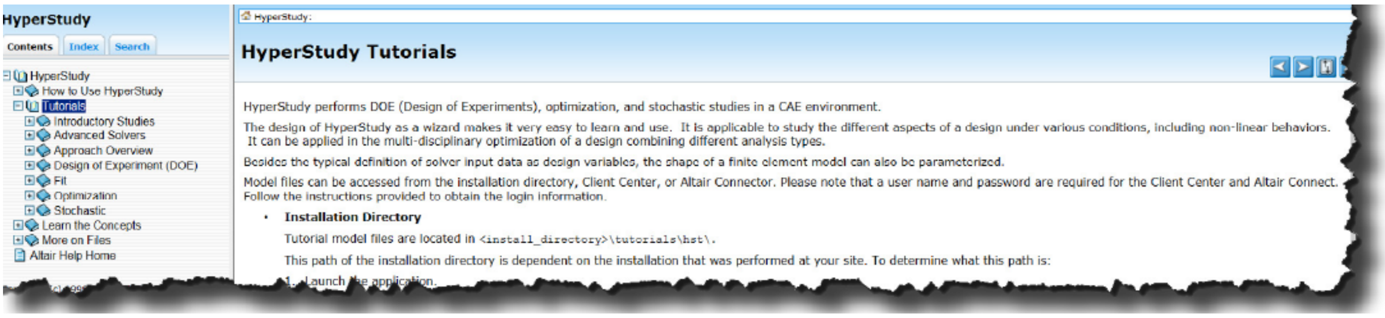
8.2 Did You Know

8.2.1 Tutorials

HyperStudy (and in general HyperWorks related) tutorials can be accessed through the HyperWorks Help Home



More than 35 tutorials are available which are grouped in Introduction, Advanced Solvers, Approach Overview, Design of Experiments (DOE), Fit, Optimization, and Stochastic.



List of Interesting Online Help Tutorials

HS-1025: Working with a HyperMesh and HyperMorph Model

http://www.altairhyperworks.com/hwhelp/Altair/hw14.0/help/hst/hst.htm?hs_1025.htm

HS-1035: Optimization Study Using an Excel Spreadsheet

http://www.altairhyperworks.com/hwhelp/Altair/hw14.0/help/hst/hst.htm?hs_1035.htm

HS-4415: Optimization Study of a Landing Beam Using Excel

http://www.altairhyperworks.com/hwhelp/Altair/hw14.0/help/hst/hst.htm?hs_4415.htm

HS-1540: Shape Optimization Study Using HyperMesh and ANSYS

http://www.altairhyperworks.com/hwhelp/Altair/hw14.0/help/hst/hst.htm?hs_1540.htm

HS-1545: DOE and Optimization Study with HyperMesh, HyperStudy, and ANSYS CFX

http://www.altairhyperworks.com/hwhelp/Altair/hw14.0/help/hst/hst.htm?hs_1545.htm

HS-1550: Shape Optimization Study Using HyperMesh and Abaqus

http://www.altairhyperworks.com/hwhelp/Altair/hw14.0/help/hst/hst.htm?hs_1550.htm

HS-1570: DOE and Optimization Study Using FLUENT

http://www.altairhyperworks.com/hwhelp/Altair/hw14.0/help/hst/hst.htm?hs_1570.htm

HS-1580: DOE Study Using CATIA

http://www.altairhyperworks.com/hwhelp/Altair/hw14.0/help/hst/hst.htm?hs_1580.htm

HS-1605: Setting Up an ANSYS Workbench Model

http://www.altairhyperworks.com/hwhelp/Altair/hw14.0/help/hst/hst.htm?hs_1605.htm

HS-1615: Setting Up a FEKO Model

http://www.altairhyperworks.com/hwhelp/Altair/hw14.0/help/hst/hst.htm?hs_1615.htm

HS-4200: Material Calibration Using System Identification (RADIOSS)

http://www.altairhyperworks.com/hwhelp/Altair/hw14.0/help/hst/hst.htm?hs_4200.htm

8.2.2 Archive

When you export a study, HyperStudy creates a file with a *.hstx extension. The following files are packaged into the .hstx file:

- Study .xml file
- Resource files
- Reference files
- hstdf and .data files
- Any other files in study directory

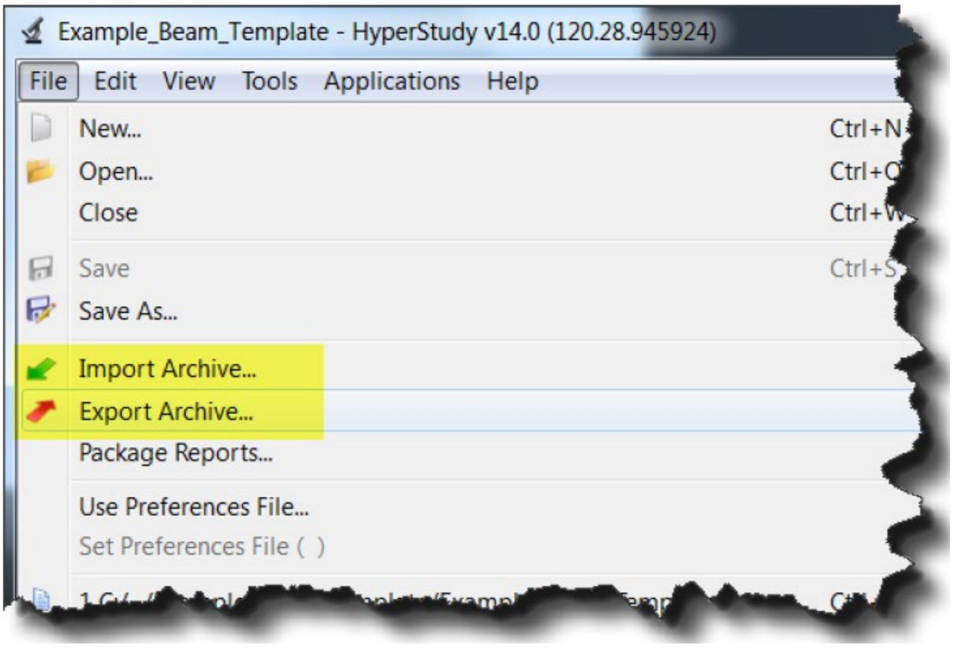
If the resource file and reference files are not in the study directory when Export Archive is selected, then they will be copied over and included in the .hstx file. The archived .xml study file will be updated to point to the new location of the resource file and reference files

Note: Export Archive does not package run folders in approach directories (such as approaches/doe_1/run_xxxx), in order to minimize the archive file size. Files included in resource files that do not reside in the study directory will not be packaged.

Use case: The Archive file it could be used by a fellow student who then would continue with your study; or the Support Team may ask you to send an Archive file so that they can understand what you have done). However, if HyperStudy was customized in some way (e.g. using a special Preference file, referencing an external solver, etc.) or run files, this information is not included in the Archive.

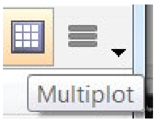
8.2.3 Import Archive

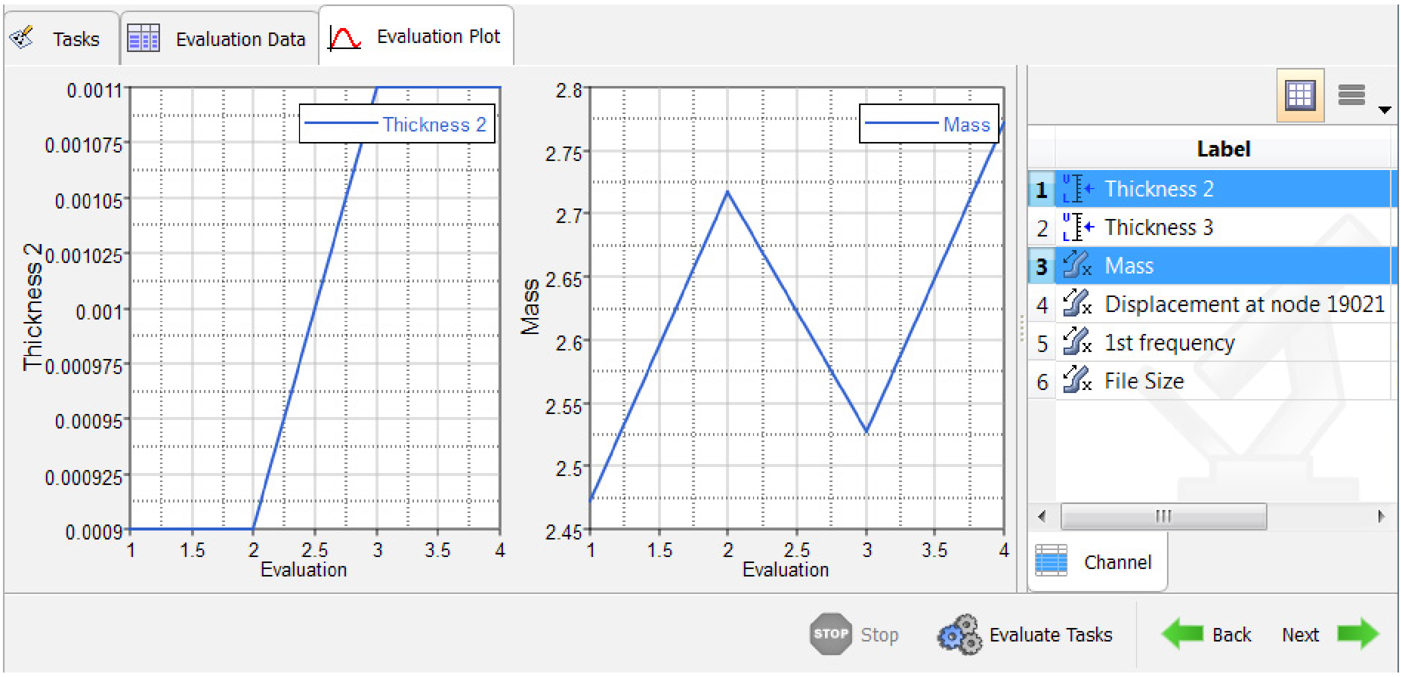
This feature un-packages the archived .hstx file at a given location. The. hstdf and .data files are placed in their respective locations and all of the other files are organized in the study directory. Import Archive also modifies the study directory in the .xml file to the new location.



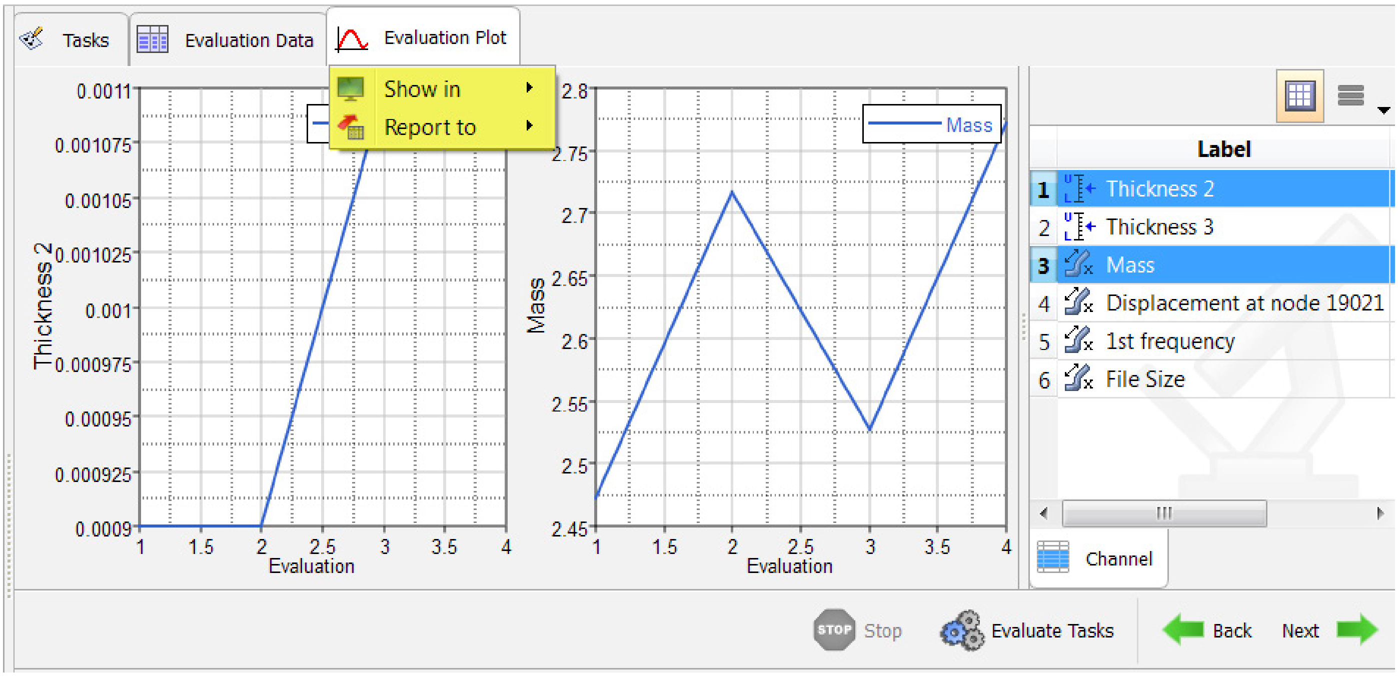
8.2.4 Multi-Plot

Activate the Multiplot icon in the upper right corner of the GUI and select the data of interest, e.g. Thickness 2 and Mass. Then HyperStudy automatically displays the 2 graphs (working area is splitted)



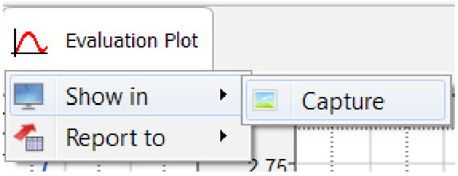


Tasks
Evaluation Data
Evaluation Plot
Thickness 2
Mass
Evaluation
Label
Thickness 2
Thickness 3
Mass
Displacement at node 19021
1st frequency
File Size
Channel
Stop
Evaluate Tasks
Back
Next



Tasks
Evaluation Data
Evaluation Plot
Show in
Report to
Mass
Evaluation
Label
Thickness 2
Thickness 3
Mass
Displacement at node 19021
1st frequency
File Size
Channel
Stop
Evaluate Tasks
Back
Next

Right mouse button clicks on "Evaluation Plot2 opens the dialog windows "Show in" → Capture (creates png file) or "Report to" → Capture (places the image in the clipboard)



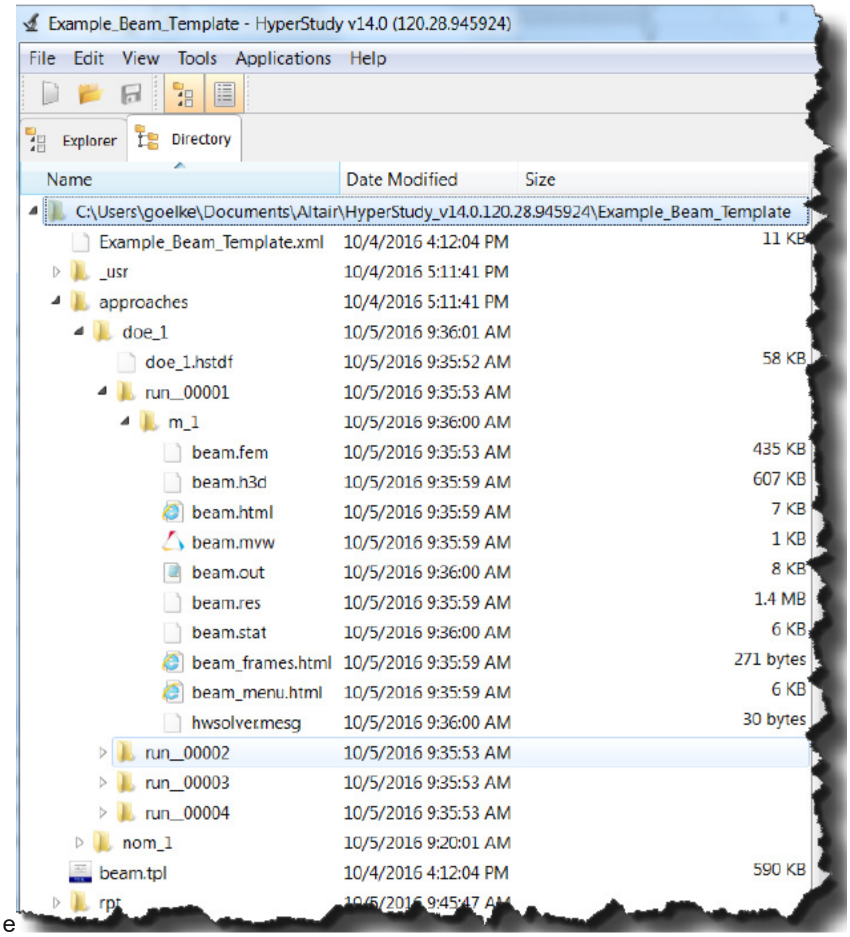
Alternatively, clicking on a graph will open the graph in HyperGraph directly.

8.2.5 Explorer

In the Explorer you can see all the studies, collapse them all (right mouse click), sort them by type, date of creation, etc.

8.2.6 Directory

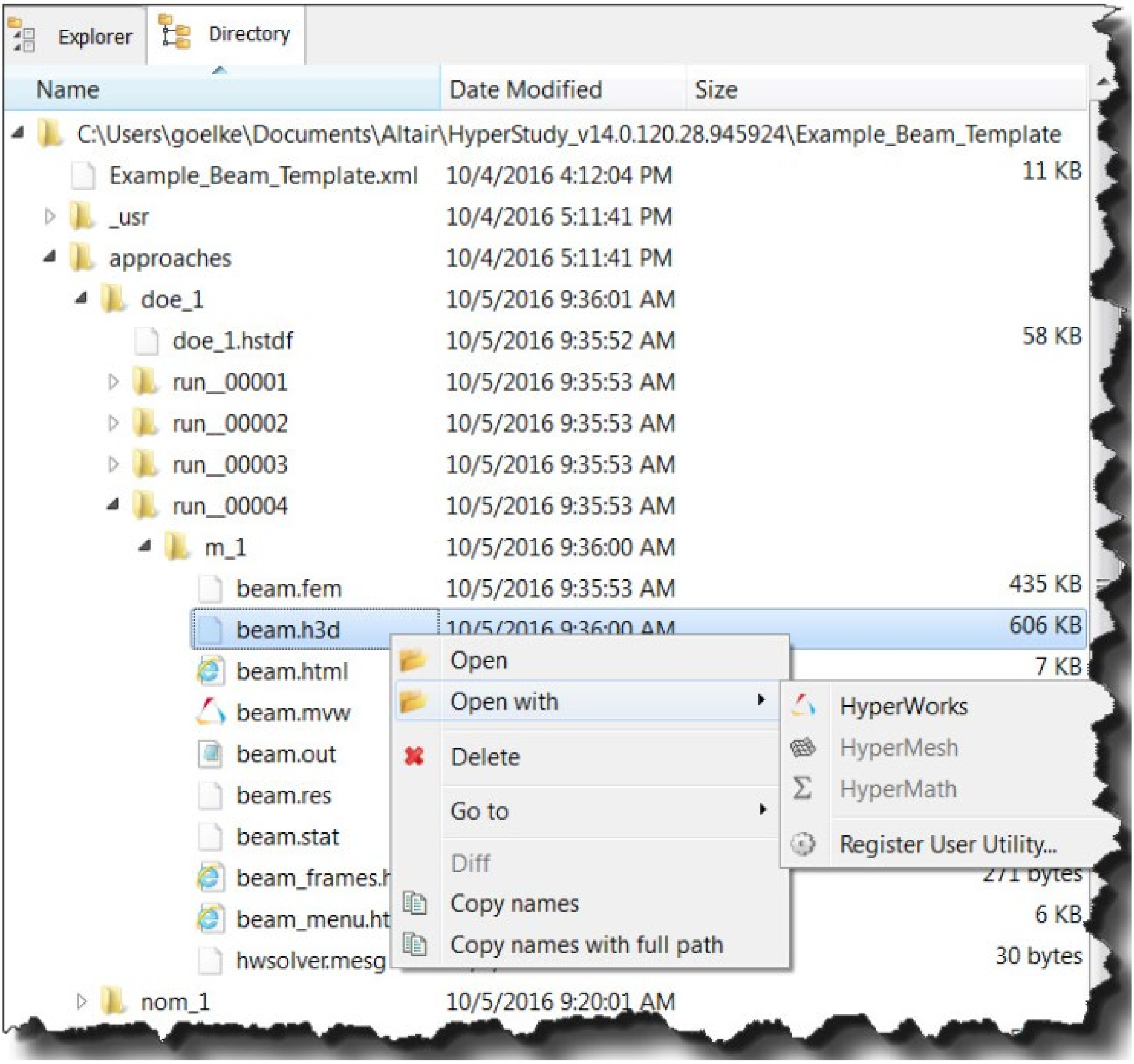
The Study directory (STUDY_DIR_PATH) is the main directory that has your xml, tpl, .fem, and other. files. It is recommended to save the preference file here as well, so when you Archive the Study, the preference file also gets archived. You can expand all the run directories (STUDY_RUN_PATH). You can "Diff" files, directories. You can open the directory in the browser. You can open h3d files in HyperView.



e

All the runs of your study will be placed under (within) the “approaches” directory. So there will also be a nominal run (nom_1) after the study setup.

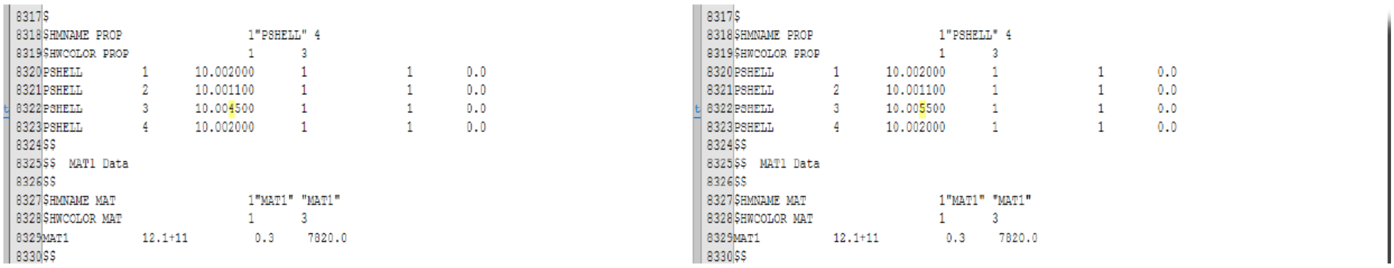
In the example above a DOE with four runs was conducted. Hence, in the directory “doe_1” there are 4 run-folders (run_0000x) belonging to the different DOE runs respectively. For instance, in the folder run_00001 the files regarding the first DOE run are saved in the folder l “m_1”. All other folders follow the same structure.



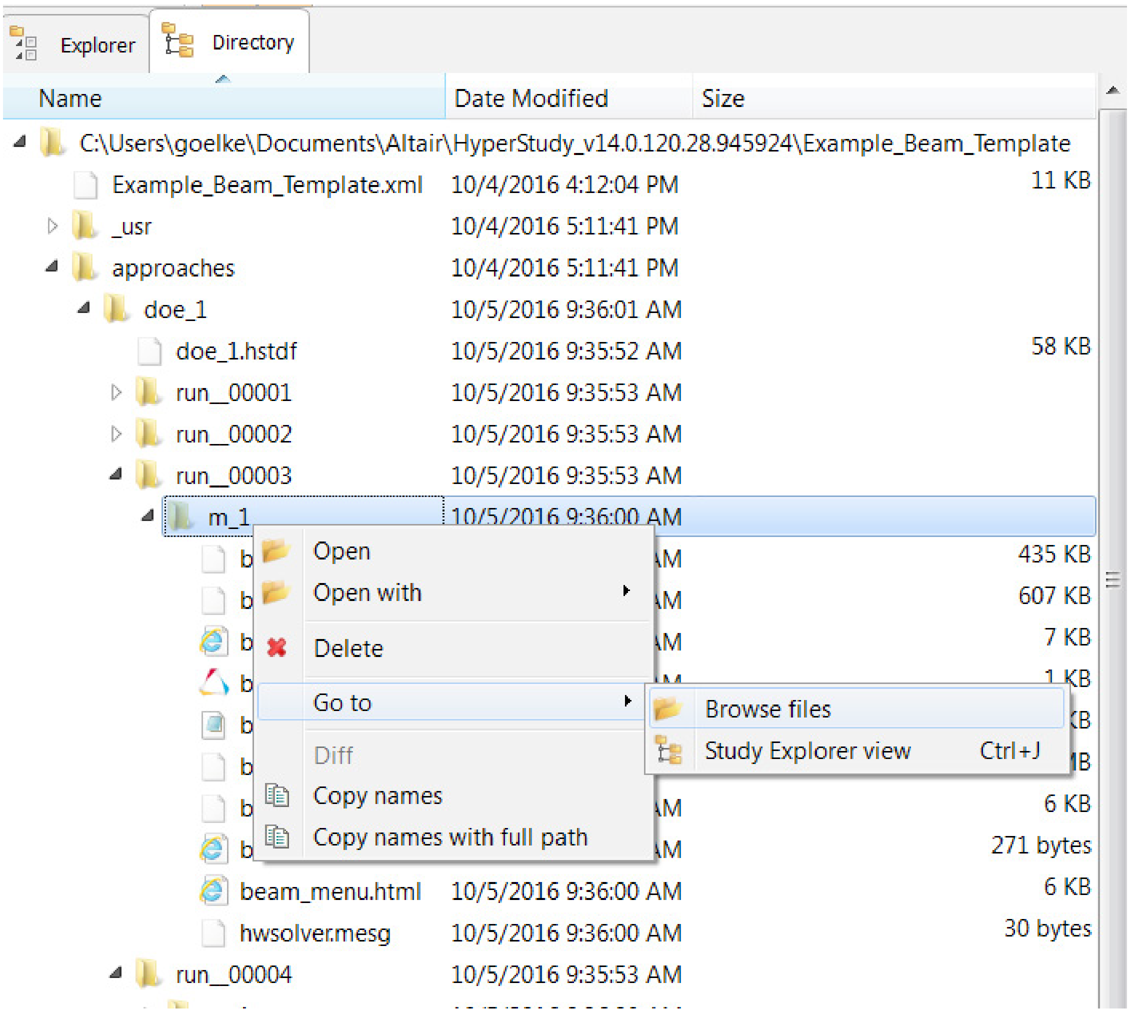
Right mouse clicks on the *.h3d file automatically loads the model and results file into HyperView.

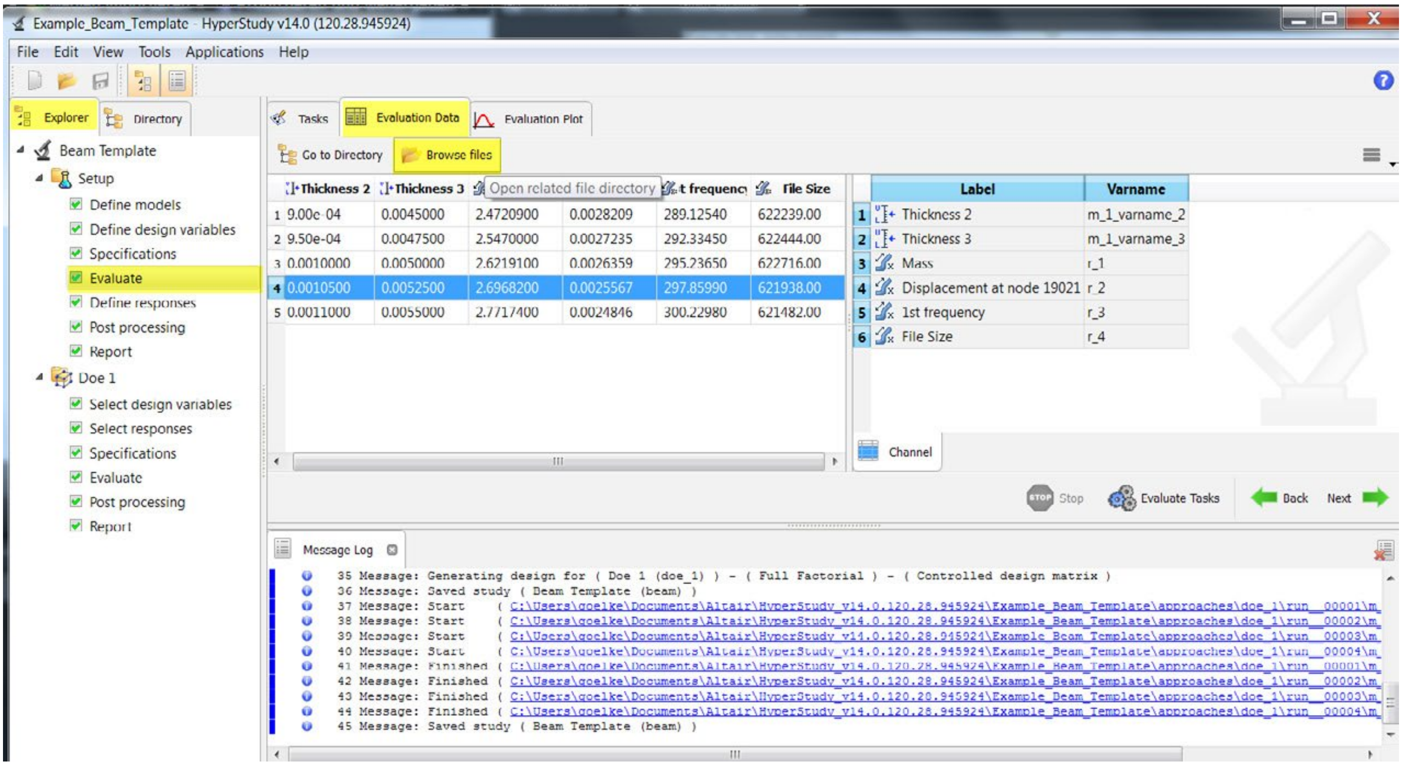
If available on your PC, you can also select run files (e.g. *.fem) from different runs/folders to compare (and maybe debug) your model. Pretty handy functionality.

Diff allows to visualize difference of the input files (model files). You may also compare (diff) entire directories. You may need to set the ENV Variable to a Diff tool (Diff-HW_HST_CMD_DIFF)



Also, very convenient is the option to “Browse files” which list your files in your Windows file browser

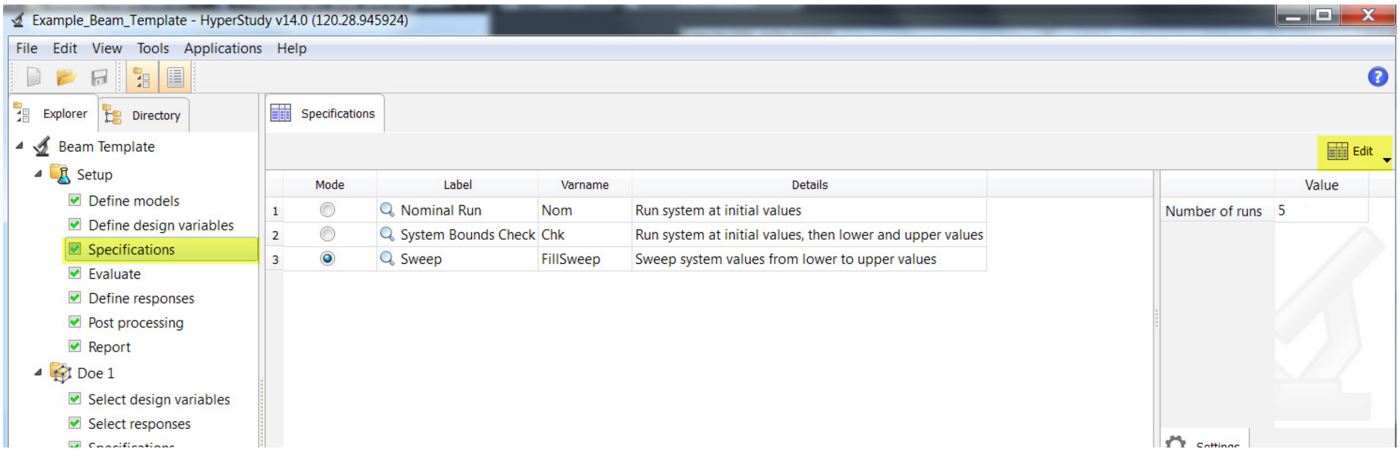




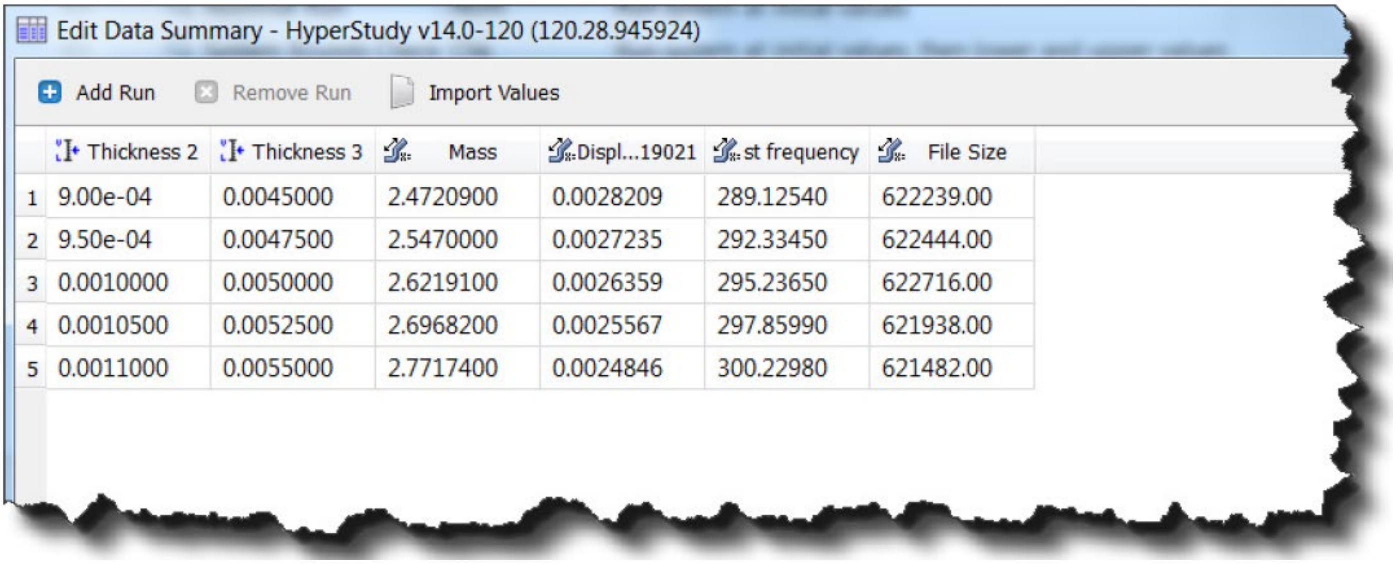
In a similar fashion, if you want to see all files belonging to a particular design, you may go to the "Explorer" →Evaluate, then select the design of interest and click on the "Browser files" option in the header. This will then open up the file explorer listing all relevant files.

8.2.7 Run Matrix

Available in the Specifications pages, top right corner (highlighted in yellow in the image below). You can add, edit, and delete runs.



After clicking on Run Matrix, the following EDITABLE summary will be shown.

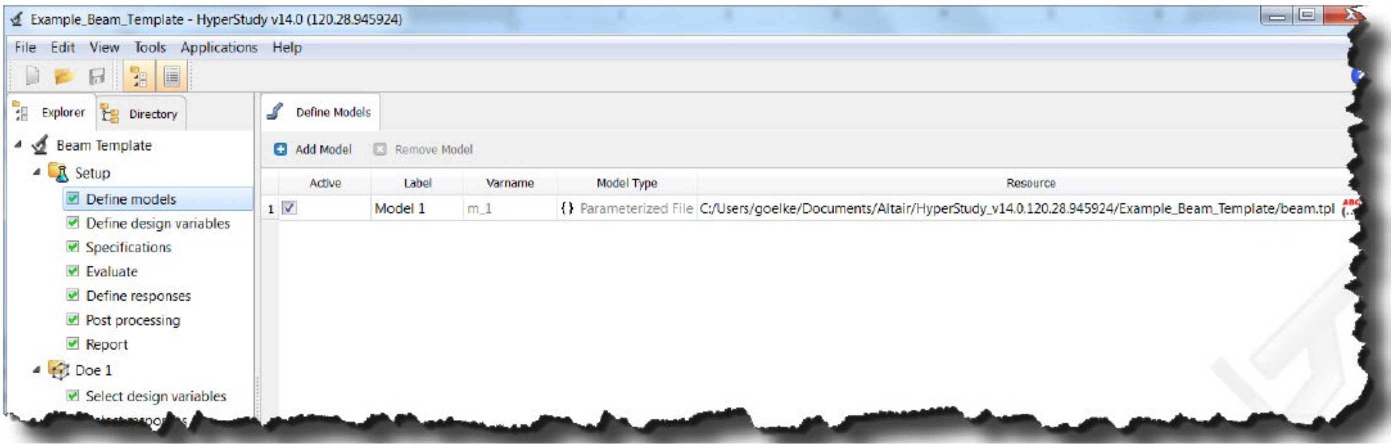


	Thickness 2	Thickness 3	Mass	Displ...19021	st frequency	File Size
1	9.00e-04	0.0045000	2.4720900	0.0028209	289.12540	622239.00
2	9.50e-04	0.0047500	2.5470000	0.0027235	292.33450	622444.00
3	0.0010000	0.0050000	2.6219100	0.0026359	295.23650	622716.00
4	0.0010500	0.0052500	2.6968200	0.0025567	297.85990	621938.00
5	0.0011000	0.0055000	2.7717400	0.0024846	300.22980	621482.00

You can modify/edit any of the fields for instance if you wish to look at your specific design "proposals". The Run Matrix offers even more: If you have test data (no simulation yet) you may include the test data in the RunMatrix (you can append more rows/columns within this table), then run a DOE, create a Response Surface on it and run an optimization study on it.

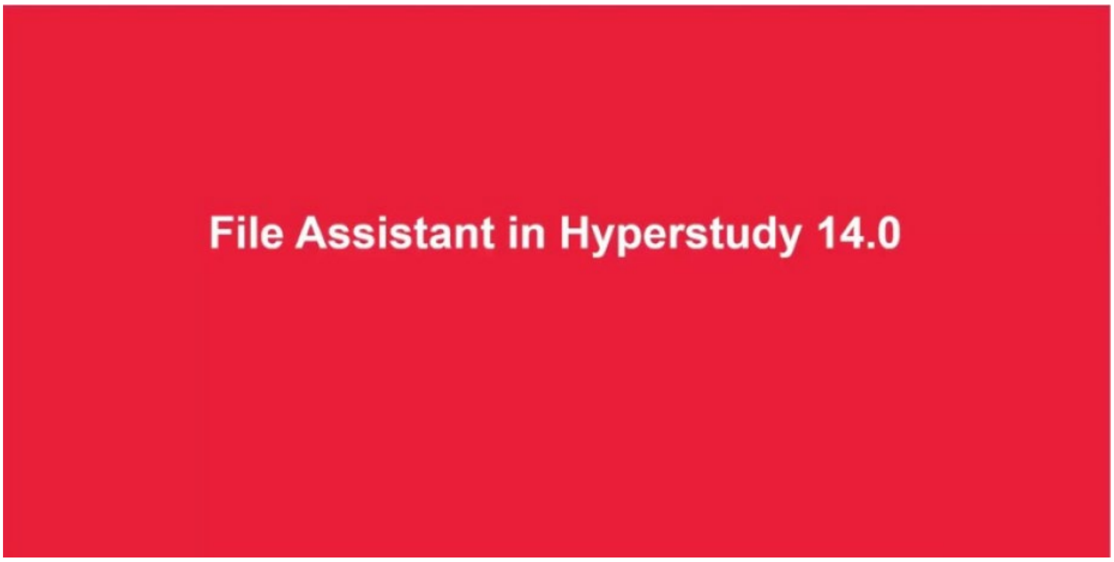
8.2.8 Drag' n Drop

While setting up your study, you can drag and drop .tpl, .hm, .xlsx, .prfj, .hstx files directly into the working area



8.2.9 File Assistant

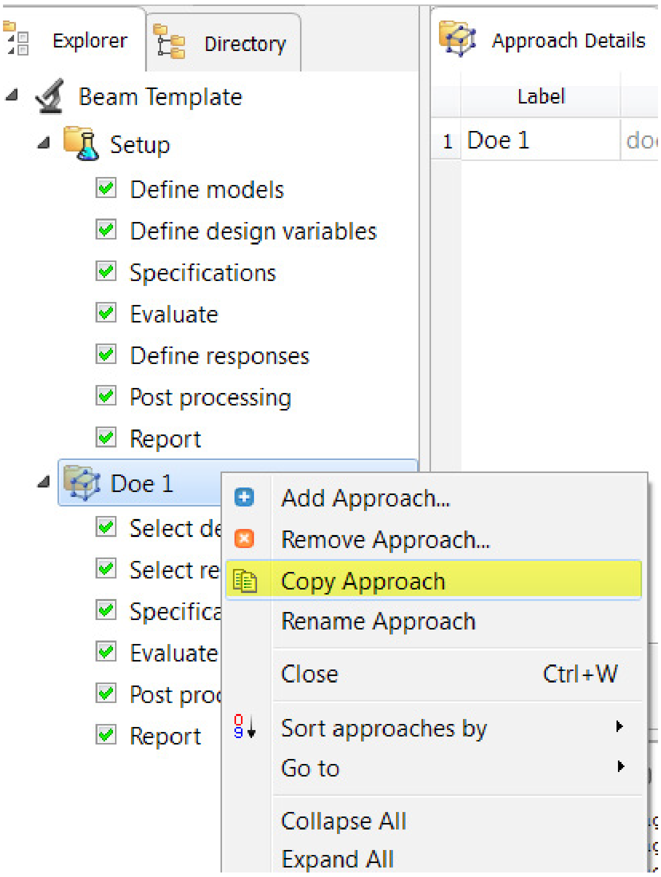
Rahul Rajan provides a brief introduction into the "File Assistant" in the video recording below.



https://altair-2.wistia.com/medias/9akrqpqpib (by Rahul Rajan)

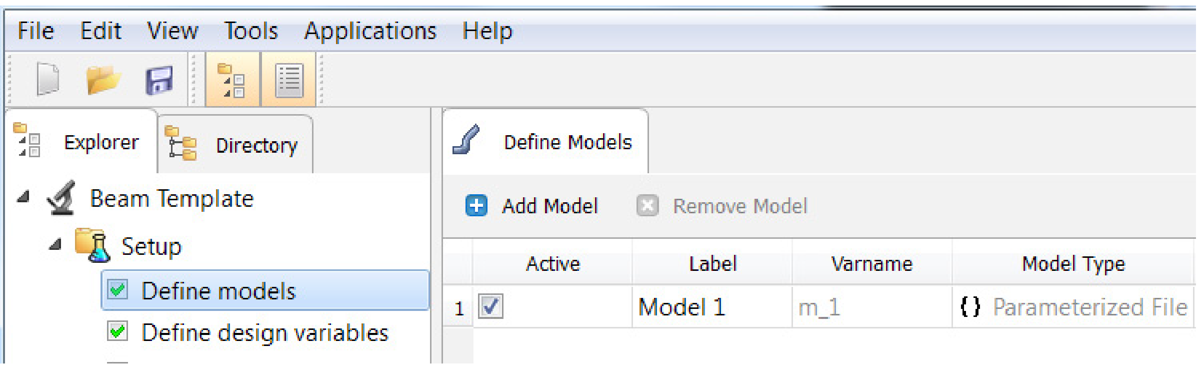
8.2.10 Copy Approaches

If you have a DOE set-up with various definitions, or you are dealing with an optimization setup with many constraints and you want to re-run it after a small modification, you can copy the approach.



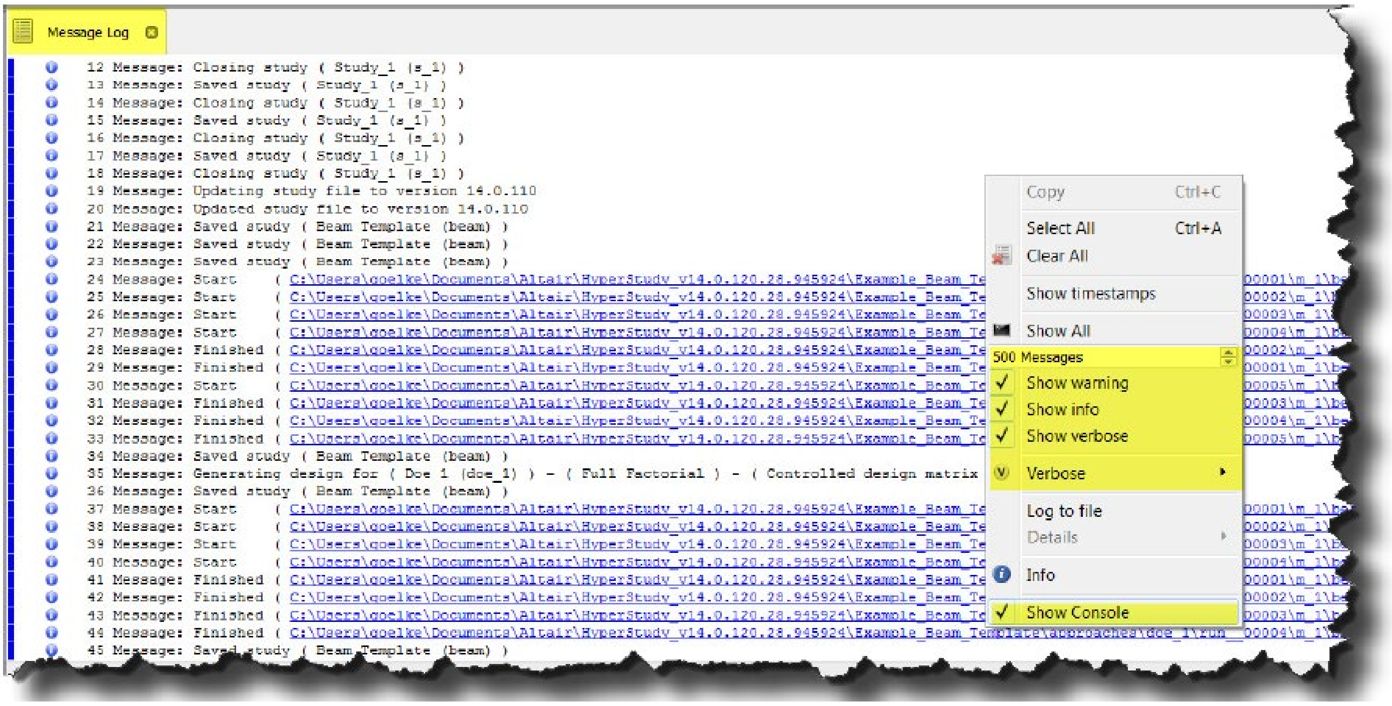
8.2.11 Varname and Label

Label can be any string, e.g. thickness for part one etc., Any characters of the supported languages should work in here. Varname is a unique identifier which is automatically assigned by HyperStudy. Of course, you can change it as well if you feel it would make your work easier. Note, you can't use the same name for two different entities.

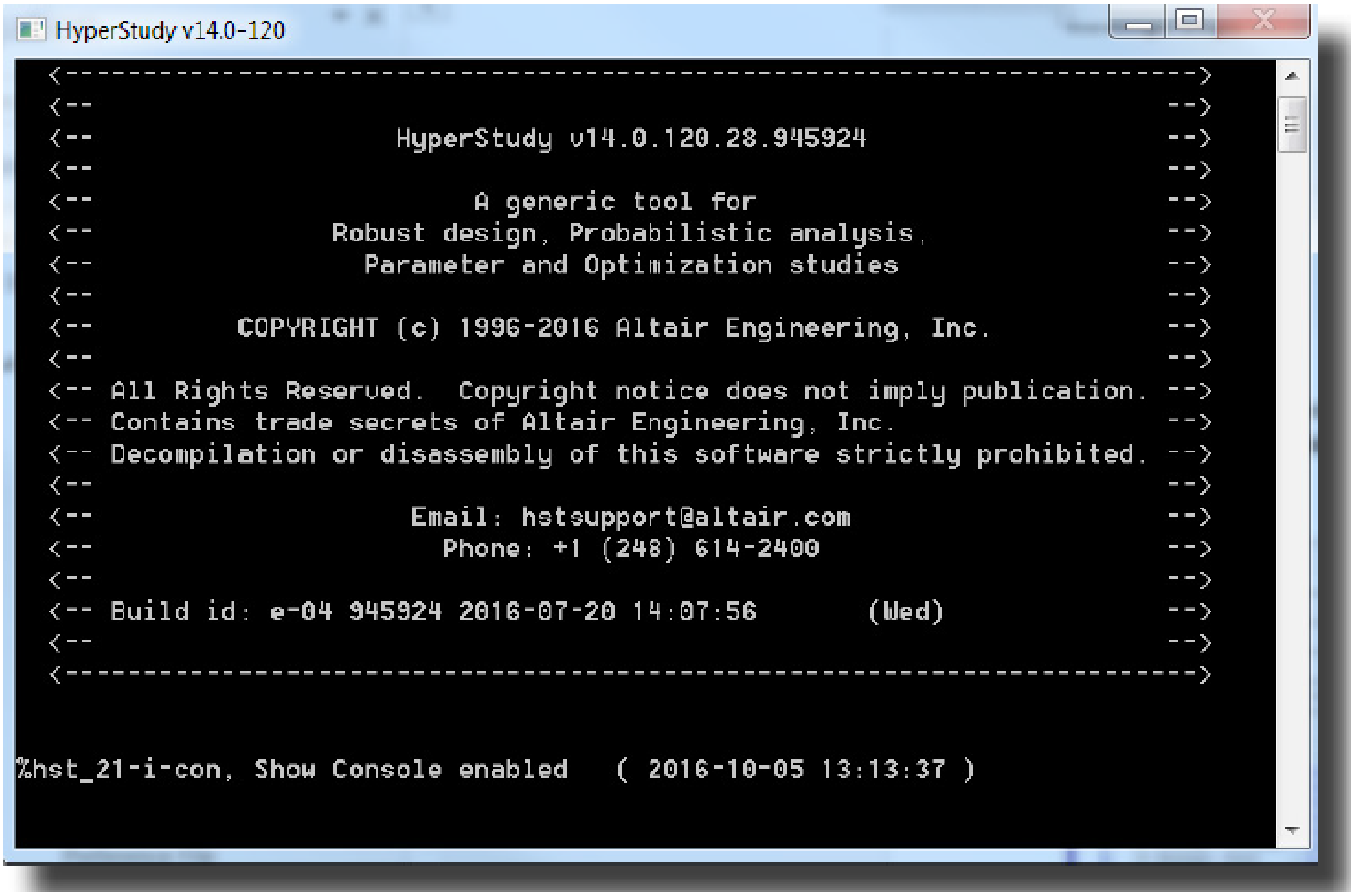


8.2.12 Message Log

The most effective way of debugging the process is to open the Console from “Message View” and to increase the verbose.



If you activate “Show Console” run information will be shown in the pop-up window:

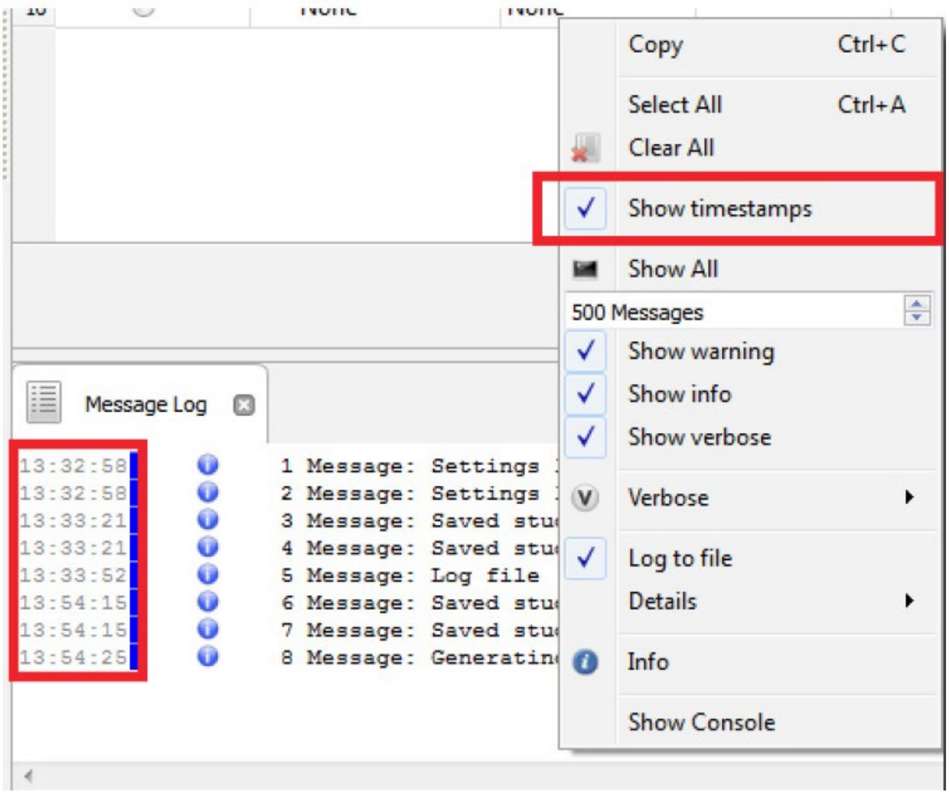


8.2.13 Time Monitoring Features

As one goes through the design exploration process it is helpful to know how long particular steps in the process are taking or when it occurred. HyperStudy logging provides two ways to monitor this. If one chooses to have a log file written during your session, a timestamp is included with each message posted to the log. For example:

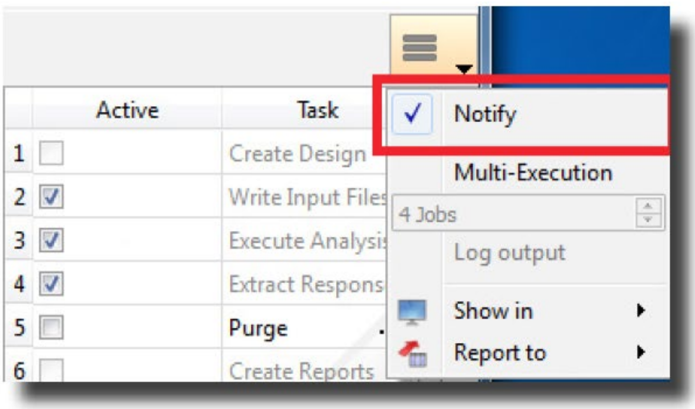
<Entity id="8" time="13:54:25" type="Info">Generating design for (Doe 1 (doe_1)) </Entity>

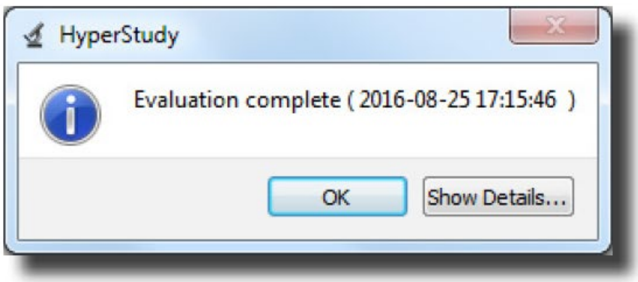
Alternately if one is monitoring the message log window one can enable the timestamps via the message log window context menu and then the message log window displays a corresponding timestamp for each message.



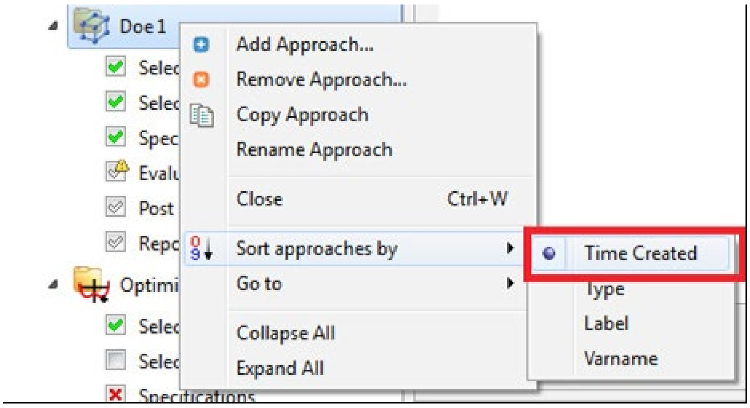
Another feature which can remind you of the evaluation completion time is to enable the ‘Notify’ dialog in the Evaluation form.

Once the evaluation is completed a dialog box will be displayed indicating the completion of the evaluations.





Yet another time reference feature is to sort the Approaches by their “Time Created” via the Study Tree context menu sort option.



8.2.14 Documentation

Here is how the HyperStudy documentation (online help) is structured:

1. How to Use HyperStudy
   - Mostly for beginners, the flow of the product is explained.
   - Invoking HyperStudy page, this section explains HyperStudy start options such as –v –s for debugging a process. This is a page even advanced users can benefit from.
2. Tutorials
   - There are 53 tutorials in HyperStudy Online Help!!! (click for the tutorial map)
   - Some show the steps to perform a task such as HS-1020: Working with a Parameterized File Model for Size Variables or HS-1060: Linking Variables of a Model to Responses of Other Models.
   - Some show realistic engineering applications such as HS-4200: Material Calibration Using System Identification or HS-4415: Optimization Study of a Landing Beam Using Excel.
   - They all have a similar flow making them easy to follow.
3. Learn the Concepts
   - Deep dive into the methods.
   - You can find a method comparison table for each approach in this chapter (click for DOE method comparison table).
   - You can also find the definitions, usability characteristics and settings of each method (click for GRSM)
4. More on Files
   - Because most HyperStudy functionality is customized through files such as a preference file; file structures are explained in this chapter.
   - This is where you can also find all the process environment variables along with solver script file examples.

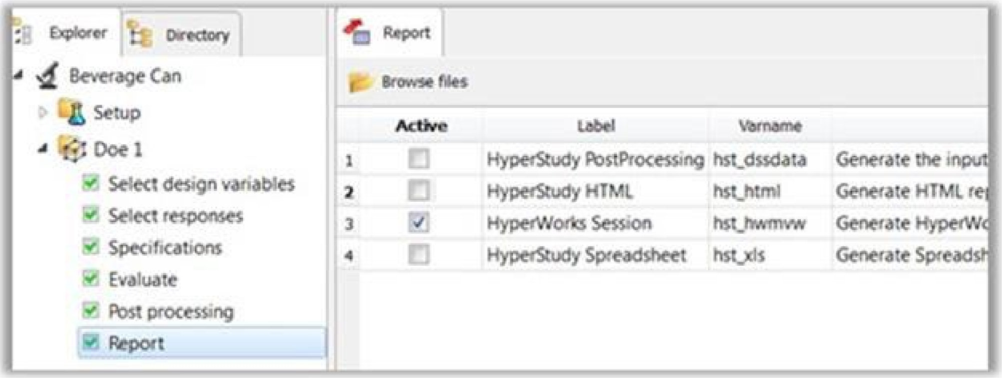
8.2.15 Reporting Capabilities & Browse Files

The typical study can produce lots of data. Of course, HyperStudy provides nice set of post-processing features to understand this data but there are times when you need to send this data to someone who doesn't have access to HyperStudy. For that HyperStudy provides some reporting capabilities.

'HyperStudy HTML', Spreadsheet and 'HyperWorks Session' can be leveraged for this particular scenario. Once you are done with report generation for all your approaches you can conveniently package reports using menu 'File -> Package Reports'. I have attached report package exported from study used for demo videos.

One more convenience feature is 'Browse files' button on Report panel as shown in image below.

This button opens folder where all your reports are stored under study directory. It comes handy when you want to access report files to do some refinements.



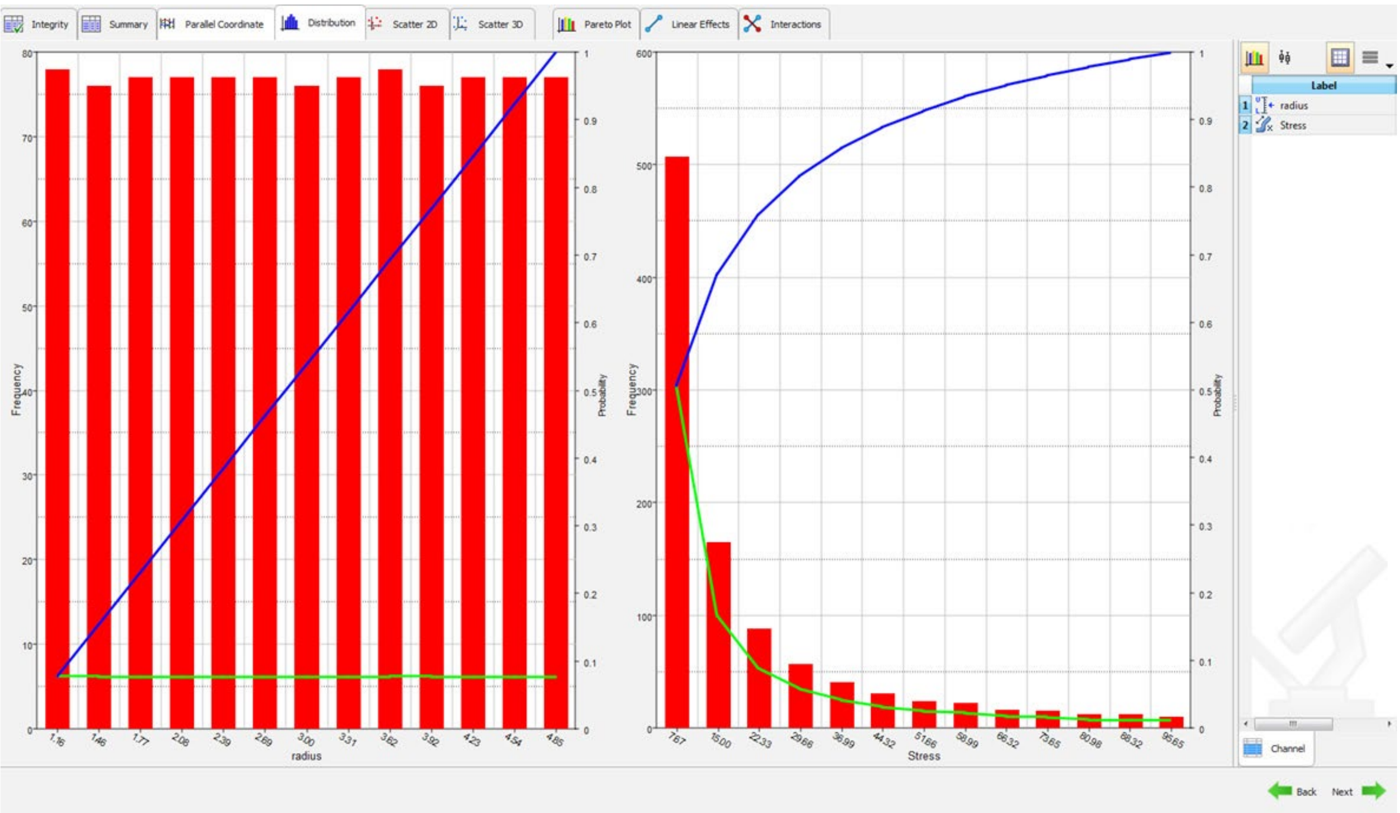
8.2.16 Statistical Distribution Terminology

Let's take a uncomplicated problem like the axial stress in a circular rod as a function of radius (stress=force/radius^2). After collecting a lot of data from a Latin Hypercube DOE into HyperStudy you see the Distribution tab (seen in the screenshot below). There are 3 related pieces of data shown the plots.

Histogram. This is the red bars that simply counts the frequency of occurrence and uses the left vertical axis. The data is grouped into equal sized bins, and a simple additive count occurs for each bin. In the plot below, you can see the radius is equally distributed between each bin (For what it's worth this is expected as the DOE is trying to equally fill the space). But the stress is not so equally distributed and shows a distinct left bias. This result itself is interesting to remember just because the variables have a particular distribution, in general, the responses do not have the same distribution.

Probability distribution function. This is the green line uses the right axis and most commonly simply called the PDF. You can imagine making this line by drawing a curve through the tops of the histogram, and then normalizing this curve so that the area under the curve is 1.0. This curve represents the likely hood of occurrence; a higher value means that value is more likely to occur.

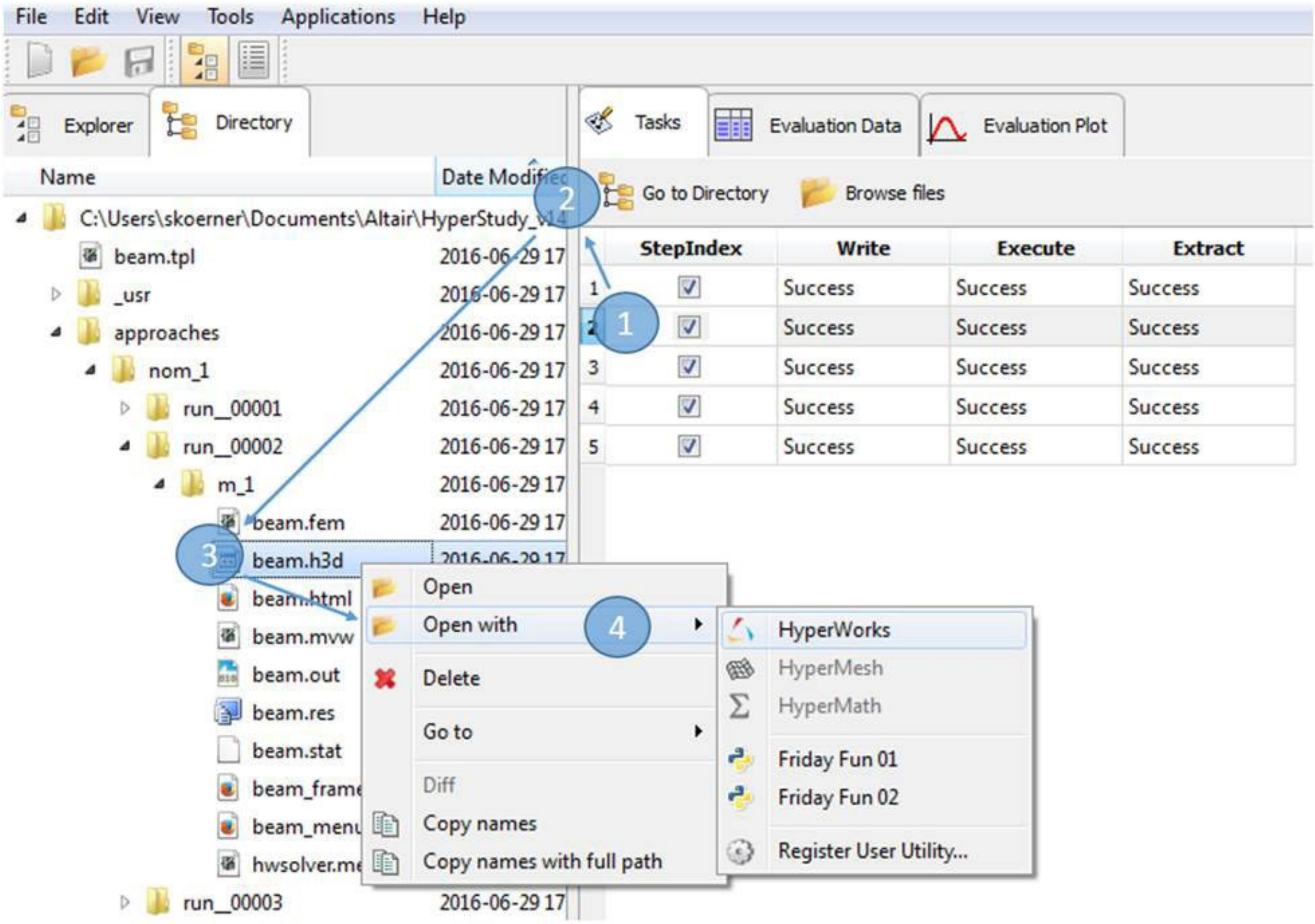
Cumulative distribution function. This is the blue curve also uses the right axis and is mostly commonly called the CDF. This curve is interpreted as the percentage of data values that falls below a given threshold. For example, 90% of the data is below a stress of 47. The normalization of the PDF is vital for this interpretation: 100% of the data must lie below the maximum value!



8.2.17 Find Result Files

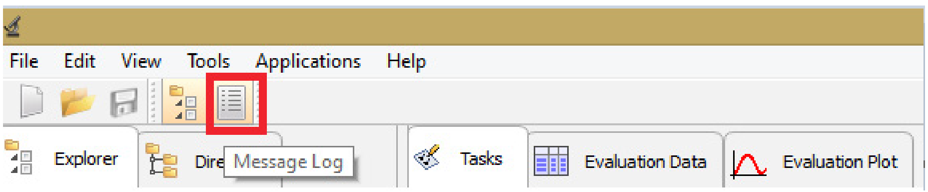
After running a study and then reopening that study a day or two later, have you tried remembering how the results of a particular run looked. Instead of fishing around on the file system to find the result files, try the following steps inside the HyperStudy application.

- In the Evaluation 'Tasks' table or the 'Evaluation Data' table, First highlight the row corresponding to the run of interest.
- Now Click on the "Go to Directory" button at the top of the form.(This will open the Directory view and the run folder corresponding to that row of the table)
- Right click on the appropriate file in the 'Directory' view and this will open the context menu with a number of options.(The standard operations can be supplemented by registering your own user utilities if you have special tools which should operate on the files)

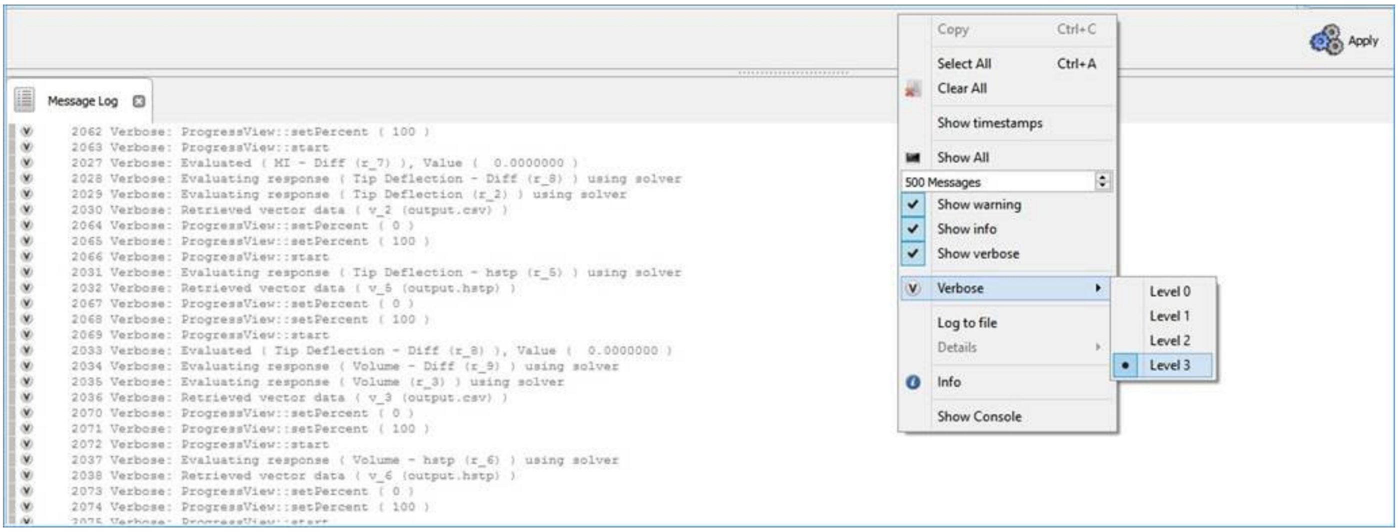


8.2.18 Verbose

Once in a while you must have encountered some kind of error or warning and looked at the lower half of HyperStudy workspace where a terse description of that warning/error was provided. That area is the “Message log” and by default it is “ON”. In case you don’t see it, then it can be turned ON using the “Message log” option in the toolbar as shown below.



If you right click in that area, you see some options. One of them is Verbose as shown in the attached image below. By default, you would notice that the option is set to Level 0. Most of the time that is good enough for simple trouble shooting/debugging or taking a corrective action in your process. However, in case you ever wished that HyperStudy could be more explicit or reveal more information about what it was doing in the background when you hit Apply, evaluate etc. then there is a simple way of doing that. Just play around with setting the levels to 1, 2 and 3. Observe that as levels increase so does the chatter from HyperStudy. At 3, HyperStudy is most talkative.



One very useful piece of information that level 3 verbose provides is writing a set of Environment Variables in a .htm file. As show in image below, you can simply click on the link and check the Env. variables and their set value for the given study. This is useful in case you want to write a tcl script or a batch file which use the Env. Variables that HyperStudy understands.

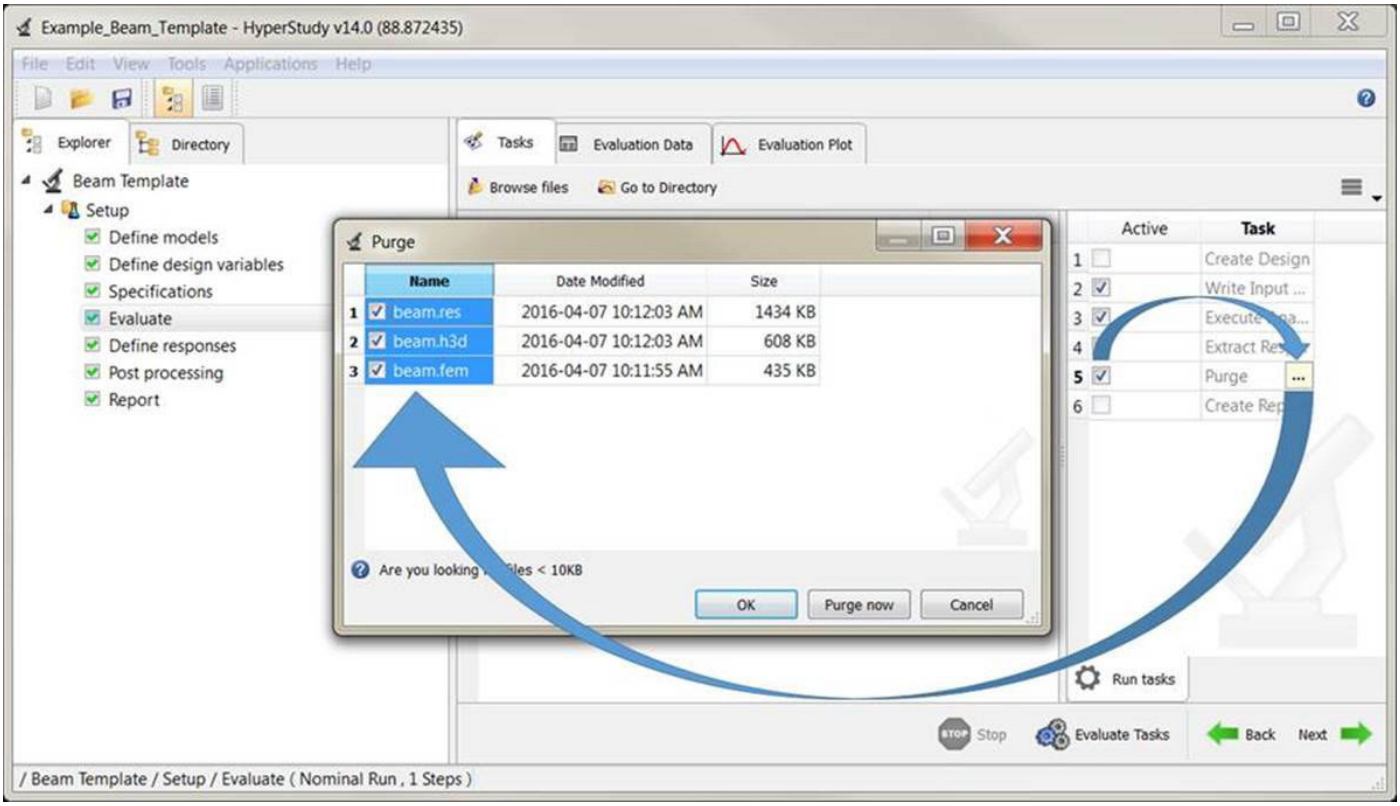
8.2.19 Purge

It is not uncommon for multi-run design exploration software, to utilize a lot of disc space and subsequently run into disc space constraints. If a single evaluation cycle uses 100 MB, 100 evaluations will have a footprint of 10 GB!

Purge is a predefined task in every evaluation cycle, that deletes hand-picked output files.

Steps need to follow:

- Finish your nominal Setup
- Activate Purge in the Evaluate Tasks list and inspect the files
- Select some or all files and confirm (We recommend hanging on to important files)
- In every evaluation cycle Purge will remove those files after the Response extraction is complete
- All subsequent Approaches and evaluations will inherent those Setup Purge settings



8.2.20 Keyboard Accelerator

Have you ever had to quickly change what was showing on your desktop?

The other day I was thoroughly enjoying myself rolling the mouse scroll wheel back and forth on the sliders in the HyperStudy Trade-off form. A very insightful activity as you can dynamically see how changes in inputs to the system influence the outputs of the system. Then I heard my boss coming who I didn't want to let know how much fun I was having, so I had to quickly change what was on my screen.

Hotkeys to the rescue.

Hit Ctrl-w, 'boom done', study was closed.

Other popular keyboard accelerators

Ctrl-d for any of the 'Work buttons'

Ctrl-n for opening a new study

Ctrl-o for opening an existing study

Ctrl-q quit the application

Ctrl-c cut

Ctrl-v paste

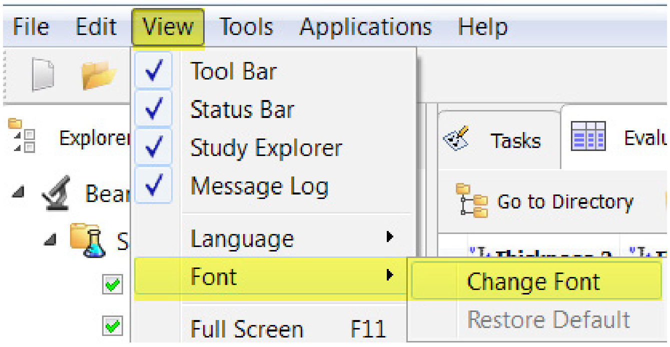
<F1> raises the help

<F11> to toggle into full screen mode and out of full screen mode (On Mac use the Mac Key instead of Ctrl)

(On Mac use the Mac Key instead of Ctrl)

8.2.21 Font

HyperStudy allows you to change the size and type of the font. Pretty nice feature.



8.2.22 Languages

The HyperStudy GUI supports different languages

Change the language by clicking View > Language from the menu bar.

English (en)

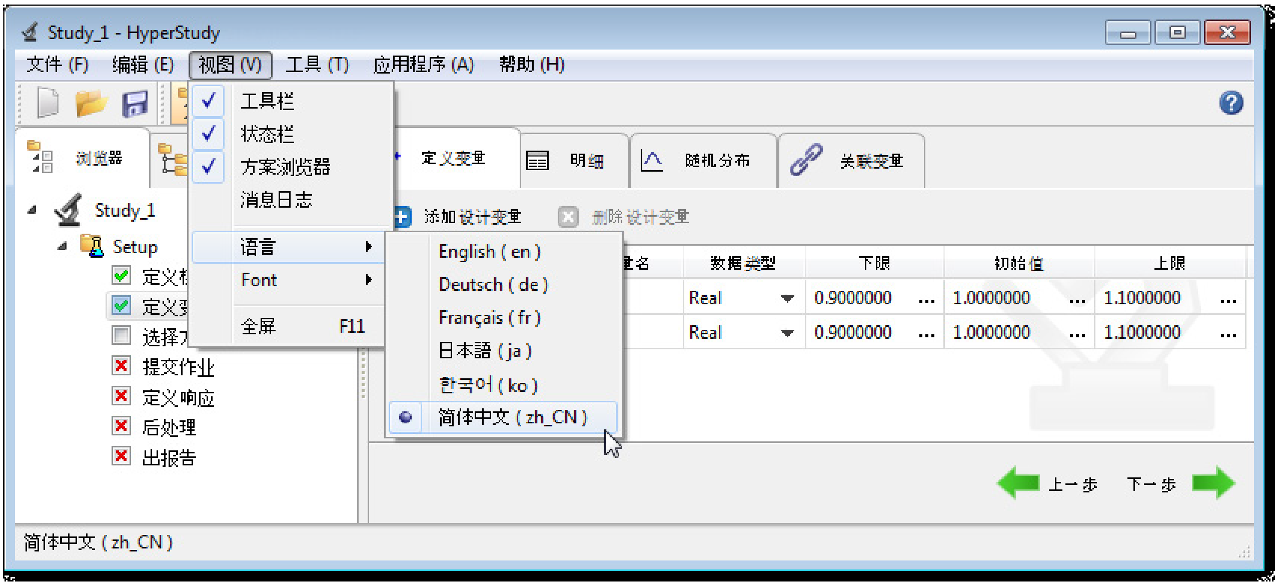
Deutsch (de)

Français (fr)

日本語 (ja)

한국인 (ko)

简体中文 (zh_CN)



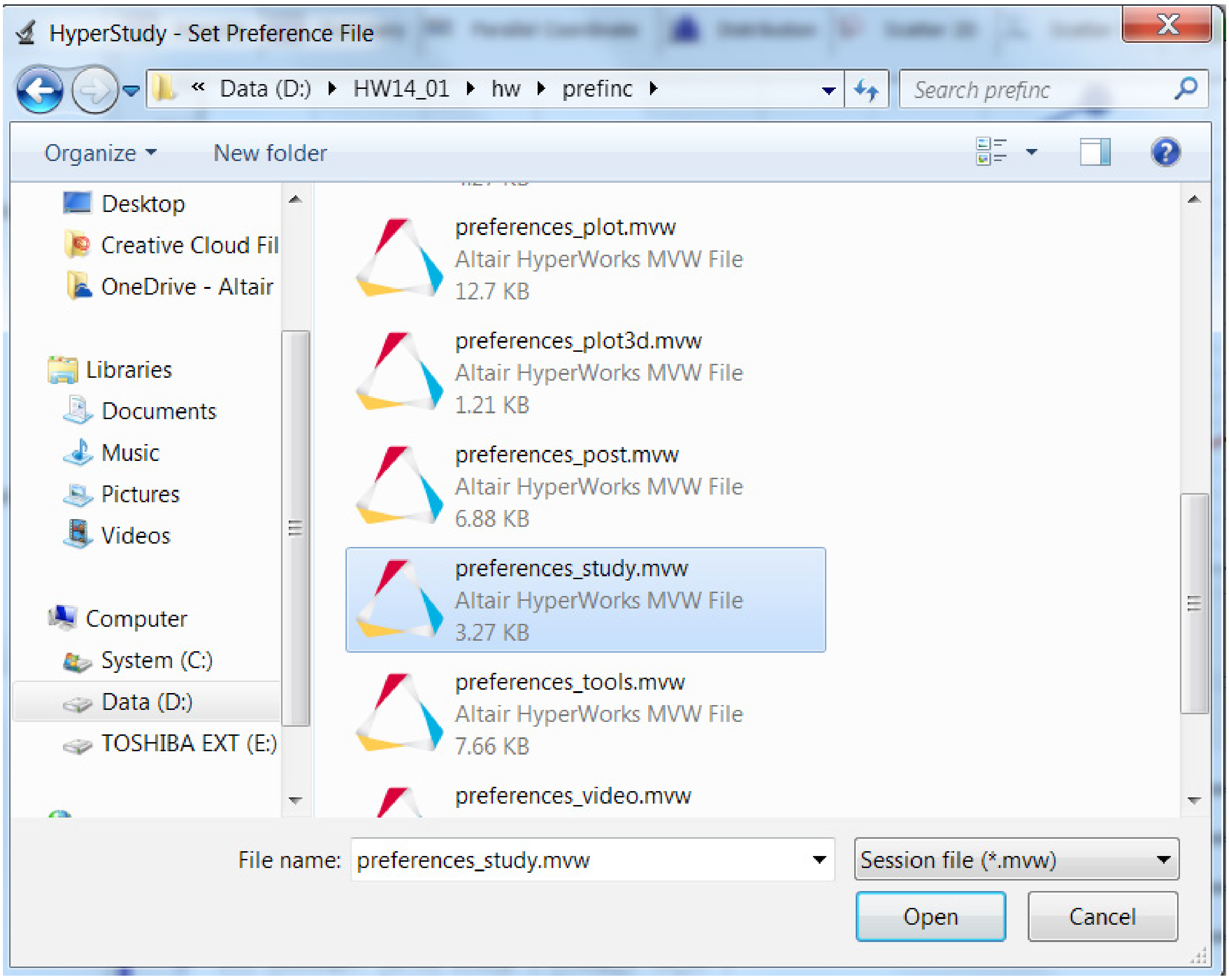
8.2.23 Preference File

You can customize HyperStudy behavior using the Preference file.

The Preference File can be registered in HyperStudy to customize studies and extend the usage of HyperStudy; such as registering external solvers, responses, optimizers, changing study precision, etc.

A preferences file is an ASCII file that configures the application or its clients, and specifies default user settings such as the readers, functions, and solver script locations. A standard preferences file is created in the program installation directory and is executed every time a license is activated. An additional preferences file can be created in your working directory, and can contain personal settings that will either overwrite the standard preferences file or be added to the existing settings in the standard preferences file.

When HyperStudy is first started, File > Use Preferences File will be inactive. In this case, only the standard preferences file ($HST_ALTAIR_HOME/hw/preferences_hst.mvw) will be used.



The default Preferences File, preferences_study.mvw in <install_dir>/hw/prefinc directory:

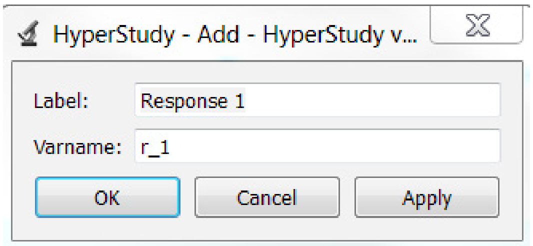
```
*BeginSolverDefaults()

    *RegisterSolverScript(radioss,     "RADIOSS",              { getenv("radioss_launch") }, HST_SolverRadioss)
    *RegisterSolverScript(os,          "OptiStruct",           { getenv("opti_launch") },    HST_SolverOptiStruct)
    *RegisterSolverScript(templex,     "Templex",              { getenv("templex_launch") }, HST_SolverGeneric)
    *RegisterSolverScript(hx,          "HyperXtrude",          { getenv("hx_launch") },      HST_SolverGeneric)

    *RegisterSolverScript(ms,          "MotionSolve - standalone", { getenv("ms_launch") }, HST_SolverMotionSolve)
    *RegisterSolverScript(py,          "Python",               { getenv("python_fullpath")}, HST_SolverGeneric)
    *RegisterSolverScript(tcl,         "TCL",                  { getenv("tclsh_fullpath") }, HST_SolverGeneric)
     RegisterSolverScript(spawn,       "HST_SolverSpawn",      { getenv("radioss_launch") }, HST_SolverSpawn)


    {if(fileinfo(getenv("hmath_batchlaunch"),"size") != -1)}
        *RegisterSolverScript(hmath, "HyperMath", { getenv("hmath_batchlaunch") }, HST_SolverGeneric)
    {endif}


*EndSolverDefaults()
```

To use a personal preferences file, you need to select File > Use Preferences File from the menu bar. In the HyperStudy - Set Preferences File dialog specify the location of the file. Once the Preference file is selected, File Set Preferences File will include the path to the file.

8.3 Response Expression Builder

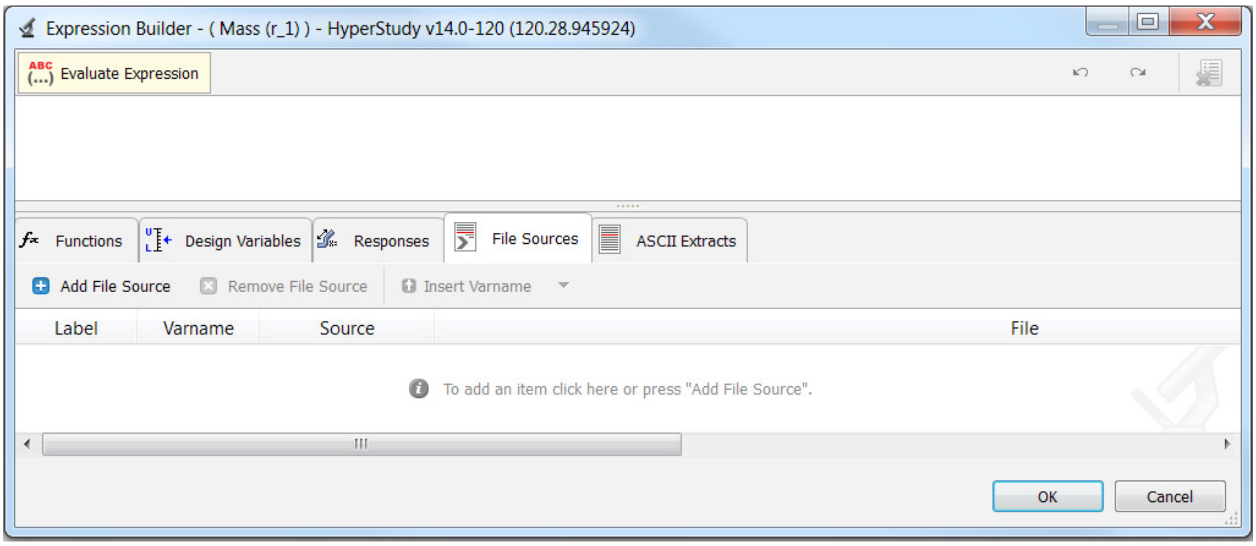
In this section the responses for the Introductory Example (Arm Model) will be set up using the Response Expression Builder. Remember that in the Introductory Example, they were created using the File Assistant. File Assistant is a UI that automates the response definition. In the background it uses the powerful Response Expression Builder. The purpose of this section is to introduce you to the Response Expression Builder so that you can use it when you need the full power of the Response Expression Builder.

Click "Add Response" to bring up the HyperStudy-Add dialog box.

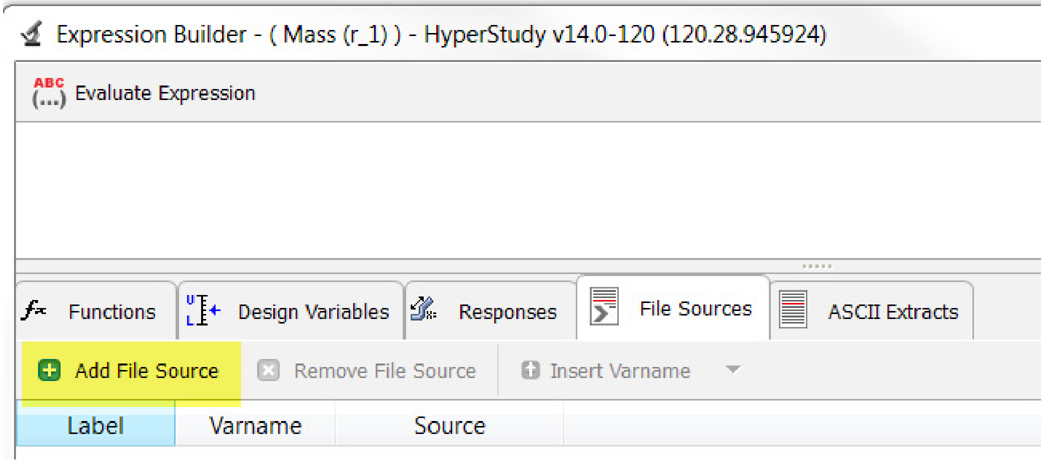


Input "Mass" as the "Label" for this response and click OK to close the dialog box.

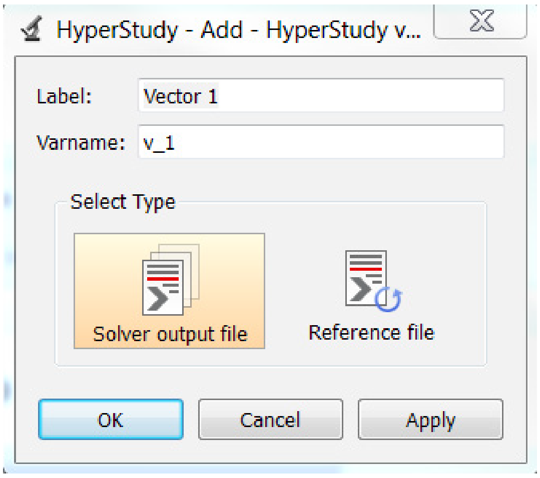
Click on "Expression …." to open the response "Expression Builder" (for the response Mass)

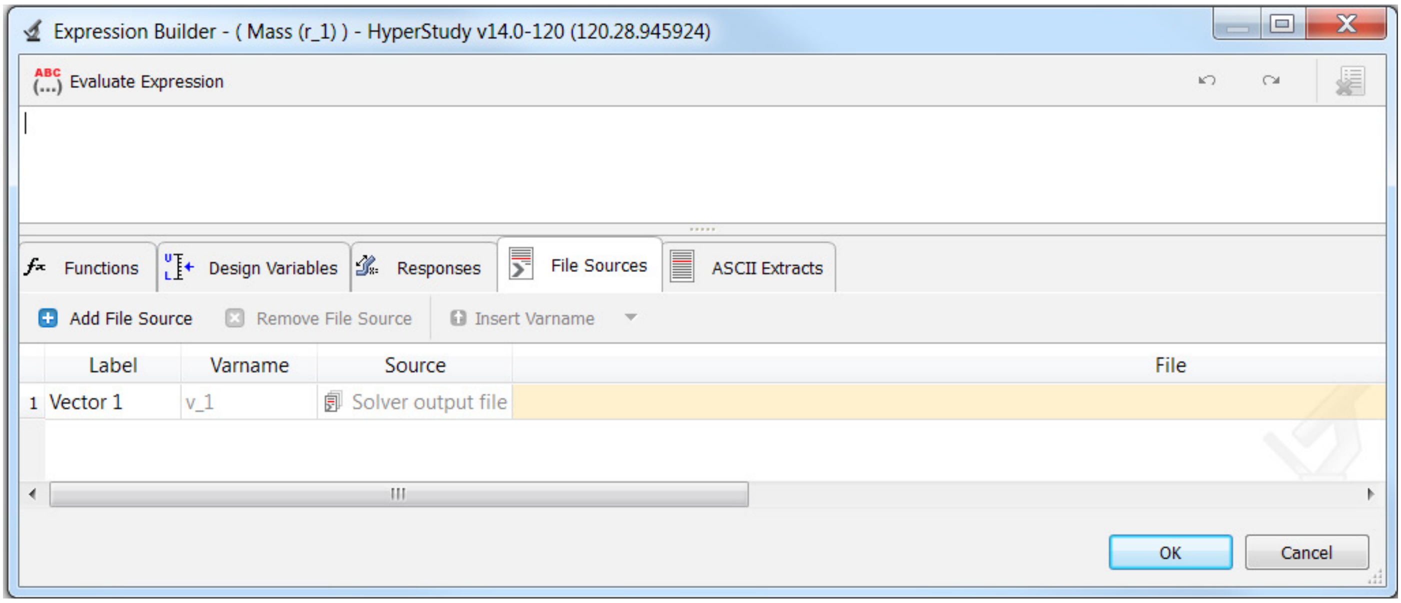


Open the “File Sources” tab (highlighted in the image above), then click on “Add File Source” to create a new vector. In fact, by specifying a “File Source” we tell the system from which file (e.g. result file, log file etc.) the respective information will be extracted from.



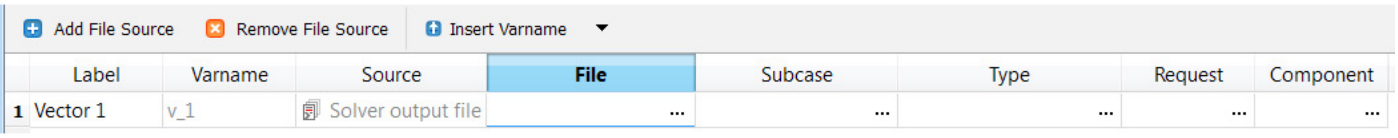
Keep “Select Type” as “Solver output file”. Click OK. “Vector 1” is created and selected in the Expression Builder.



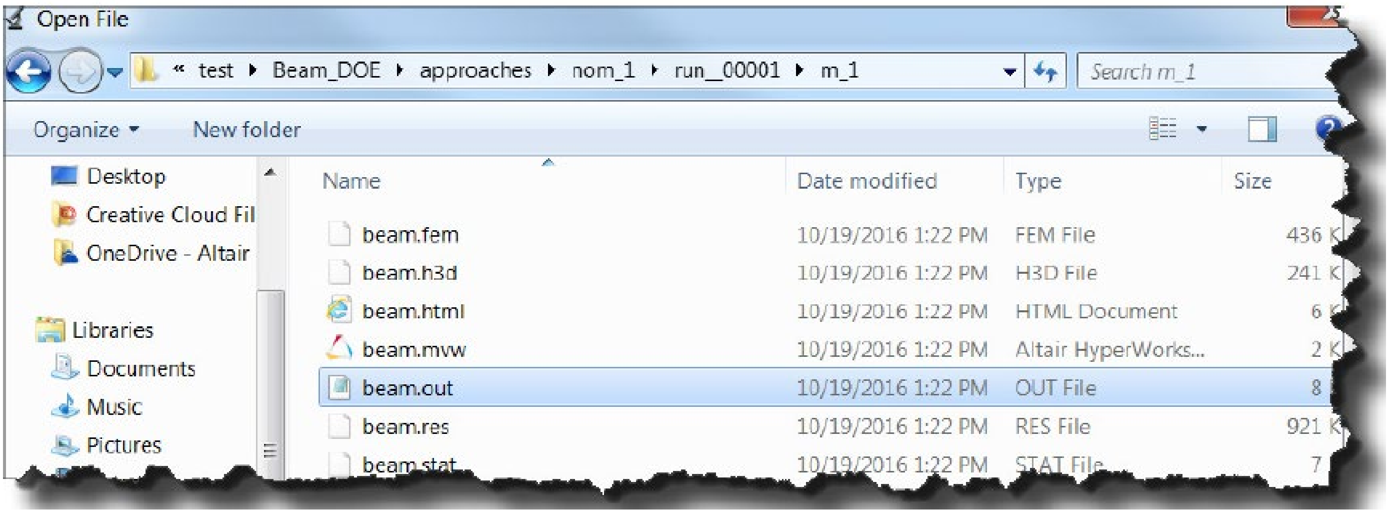


The HyperStudy Expression Builder window

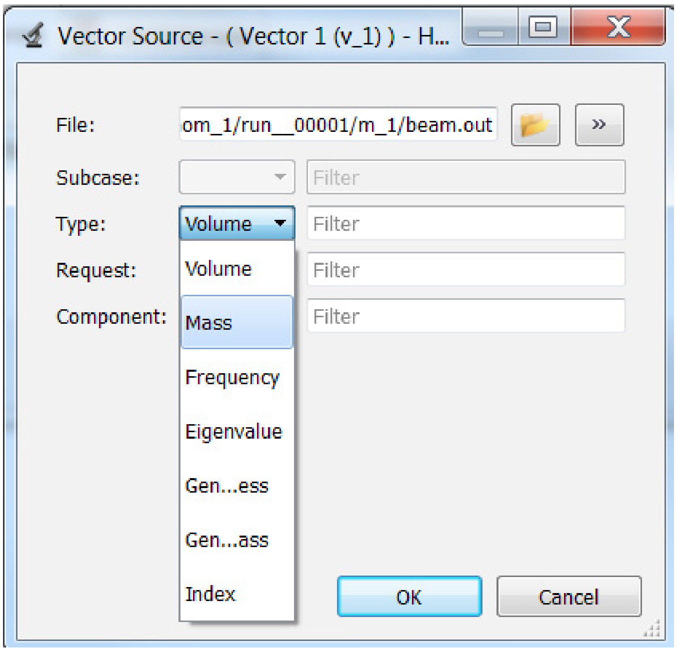
In the “File Sources” tab, click on the file icon “File …” to load a file to parse.



In the nom_run folder, open the m_1 folder and select the beam.out,



under Type select "Mass" file and click OK to load the file and close the dialog box.



Click on "Insert Varname" (image_ beam_shape_36) to add the Vector 1: v_1[0]

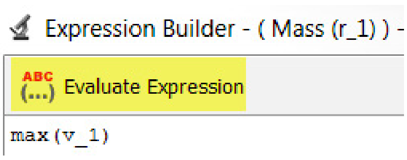
HyperStudy will read the solver output file (*.out) and assign labels to the values in the file.

Since the mass is a scalar quantity, the only component of the vector is index zero, and so a scalar mass value is properly written in the Expression Builder as v_1[0]. The mass value can be accessed directly out of the beam.out file and does not require any mathematical operations before being used as a response for our study.

Note: Multi-dimension vectors may be accessed through each individual dimension by using the corresponding index in the vector array

The Expression Builder is capable of performing a full complement of mathematical functions on an arbitrary number of vector solutions in order to build a response. For more information on the functions in the Expression Builder, access the HyperWorks help documentation for HyperStudy under Functions and Operators.

Click the "Evaluate Expression"



to display the value of the mass value extracted from the nominal run. This should evaluate to 2.16091.

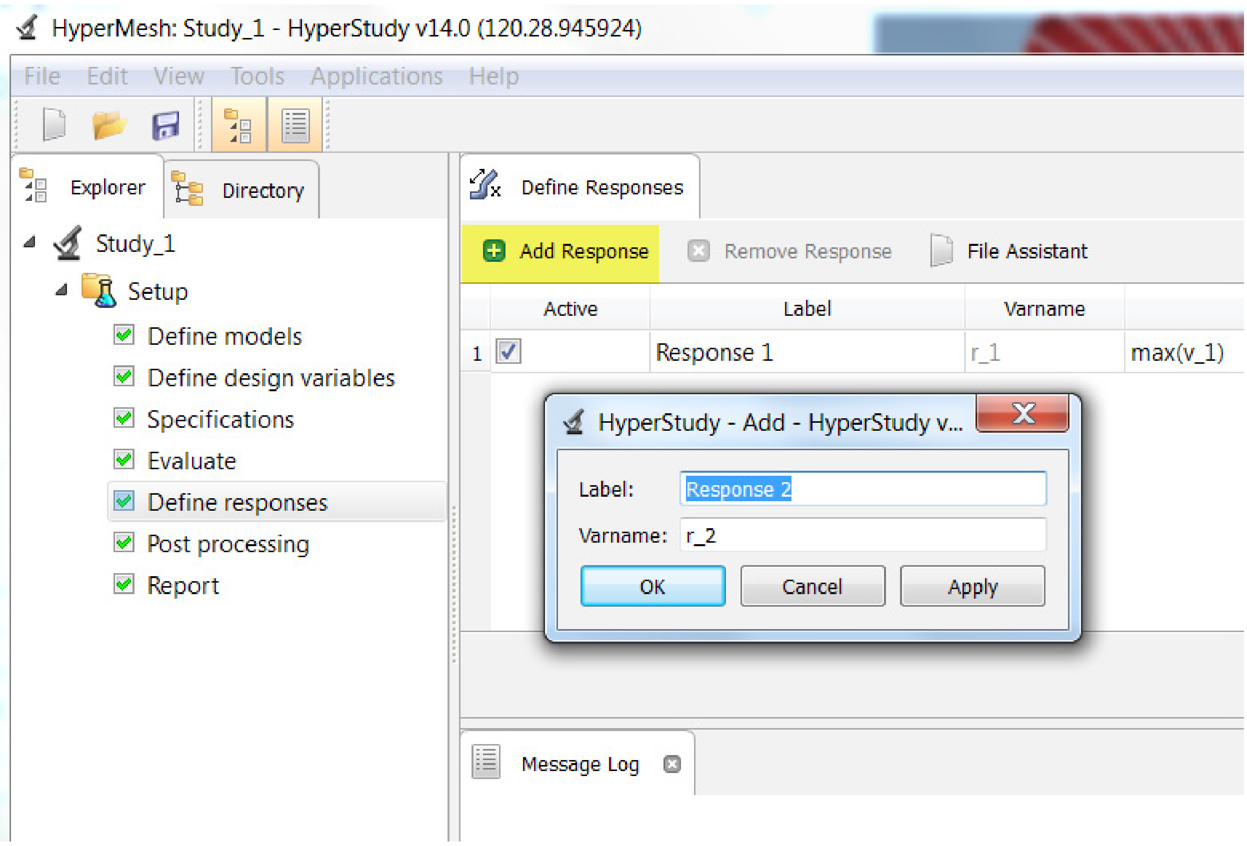
Click OK to close the Expression Builder (definition of mass as a response is completed) window and return to the Define responses step.

Repeat the previous steps to add two more responses (Displacements, Frequency) to the study according to the values below:

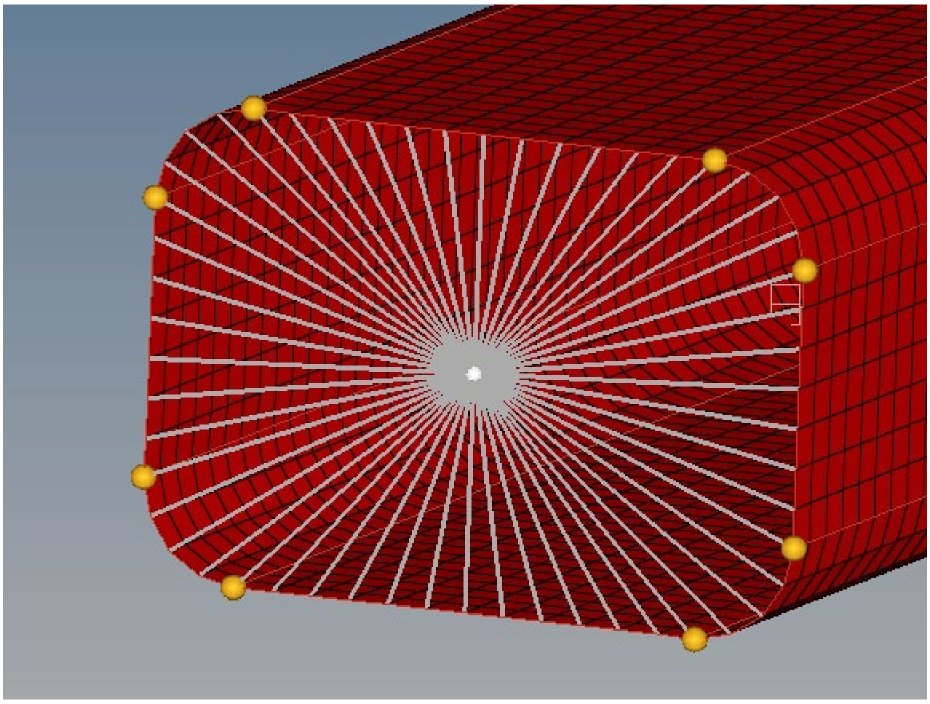
Response Name	Vector	Vector Resource File	Subcase	type	Request	Component	Evaluated expression
Displacement	Vector 2	beam.h3d	Subcase 1 (Static)	Displacement (Grids)	N19021	MAG	0.002073
Frequency	Vector 3	beam.out	(N/A)	Frequency	Mode 1	Value	368.83

Are you able to complete the next two steps by yourself already?

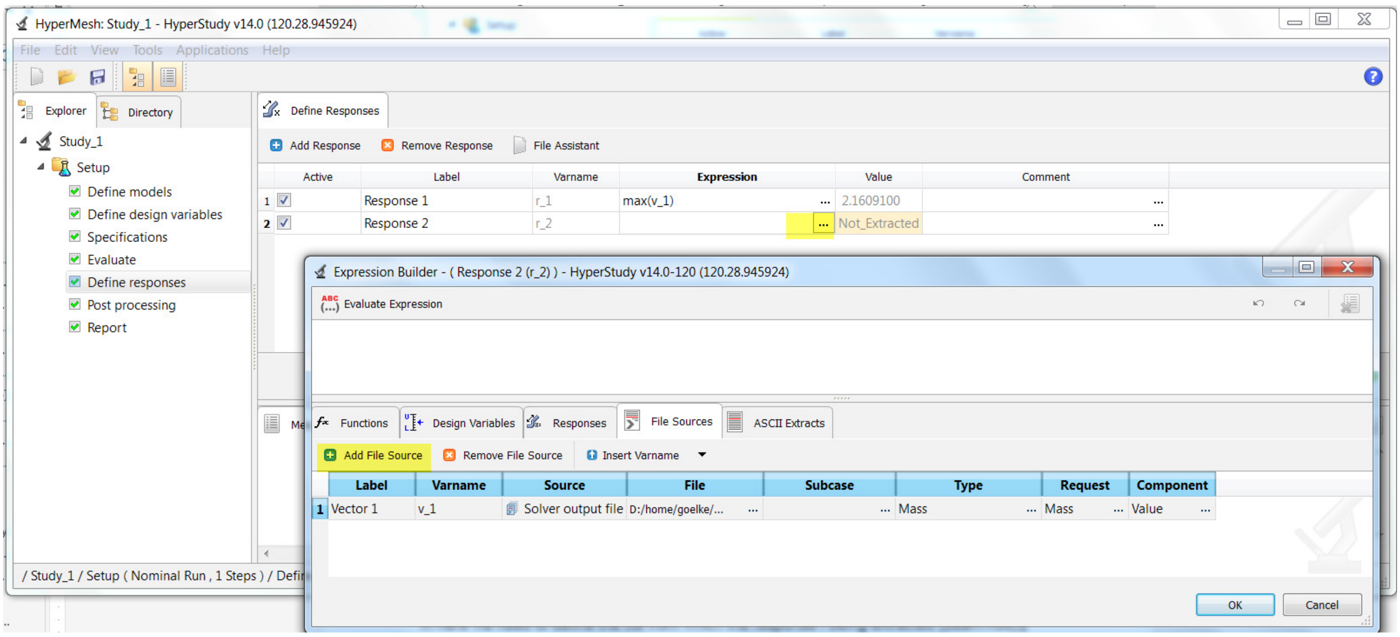
- “Add Response” named: Displacement
- Define Response of Type Displacement of Node 19021



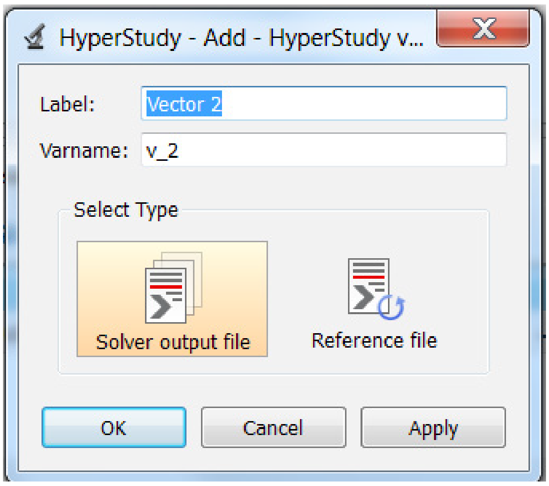
In order to define the respective vector, click on “Expression …” which opens the “Expression Builder”. Activate “Add File Source” to specify the file from which the results (displacement of node 19021) are to be extracted.

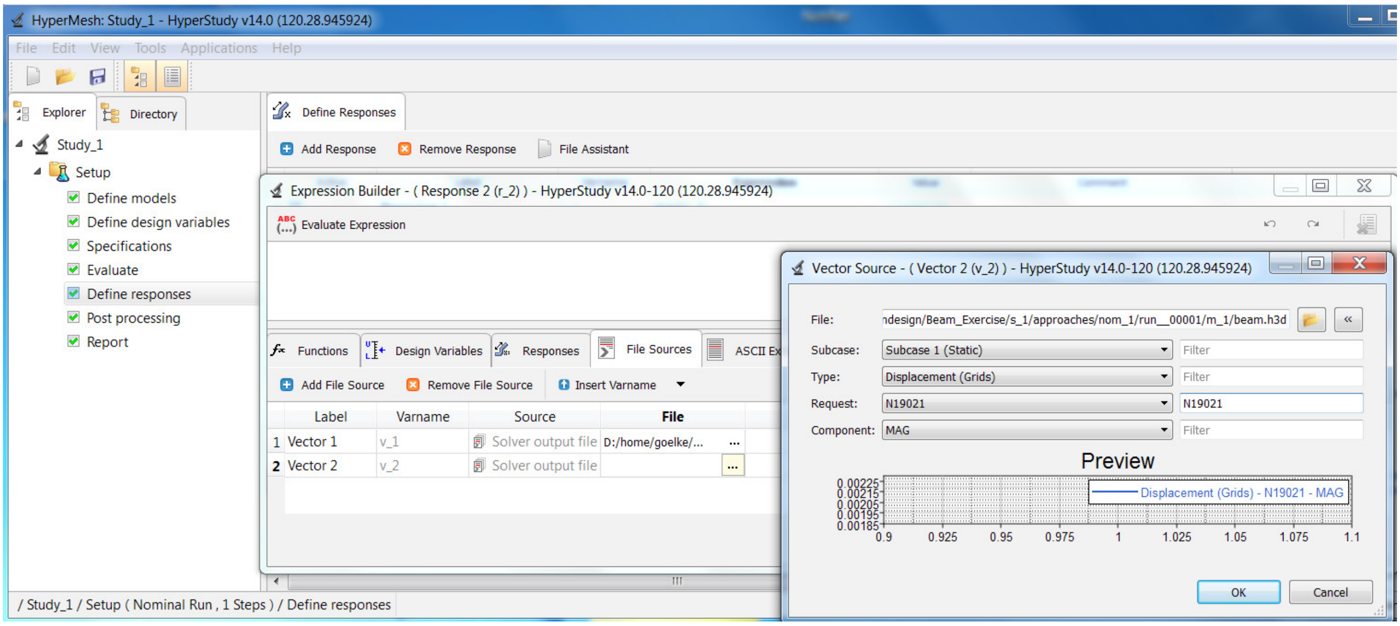


Location of node 21079



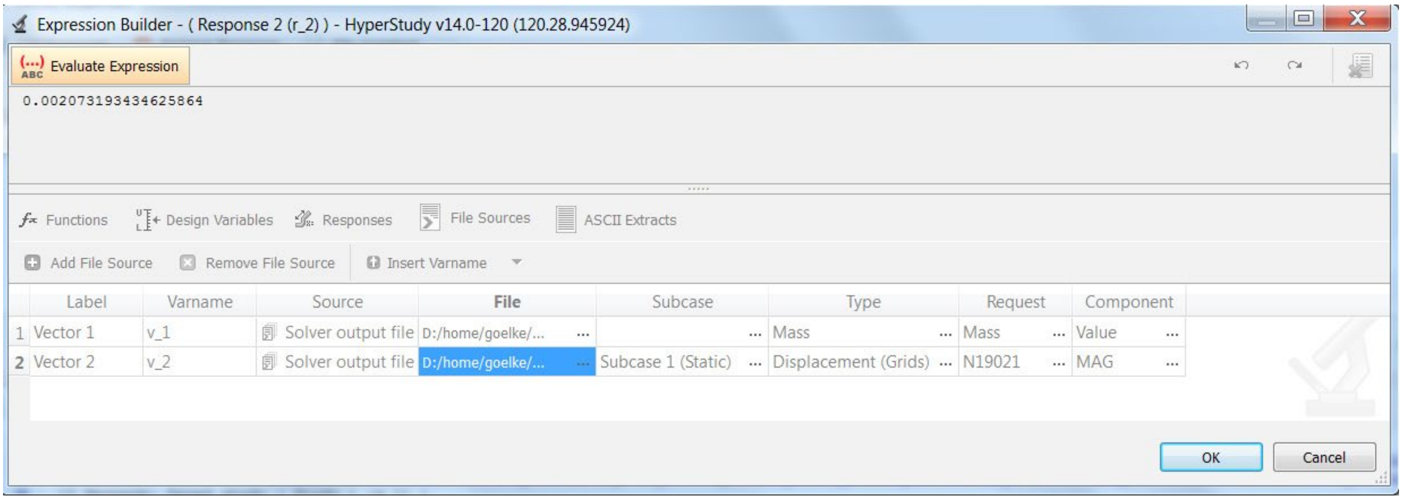
In case you are not familiar with OptiStruct, the information about displacements are contained in the beam.h3d file.





After confirming this operation by clicking OK, don't forget to activate "Insert Varname". Only then the newly created vector will be shown/listed in the panel below "Evaluate Expression".

To see whether Vector 2 was correctly defined, click on "Evaluate Expression" (if you wonder whether the displacement value displayed is correct at all – you can postprocess the results of the nominal run like any other simulation run in HyperView first).

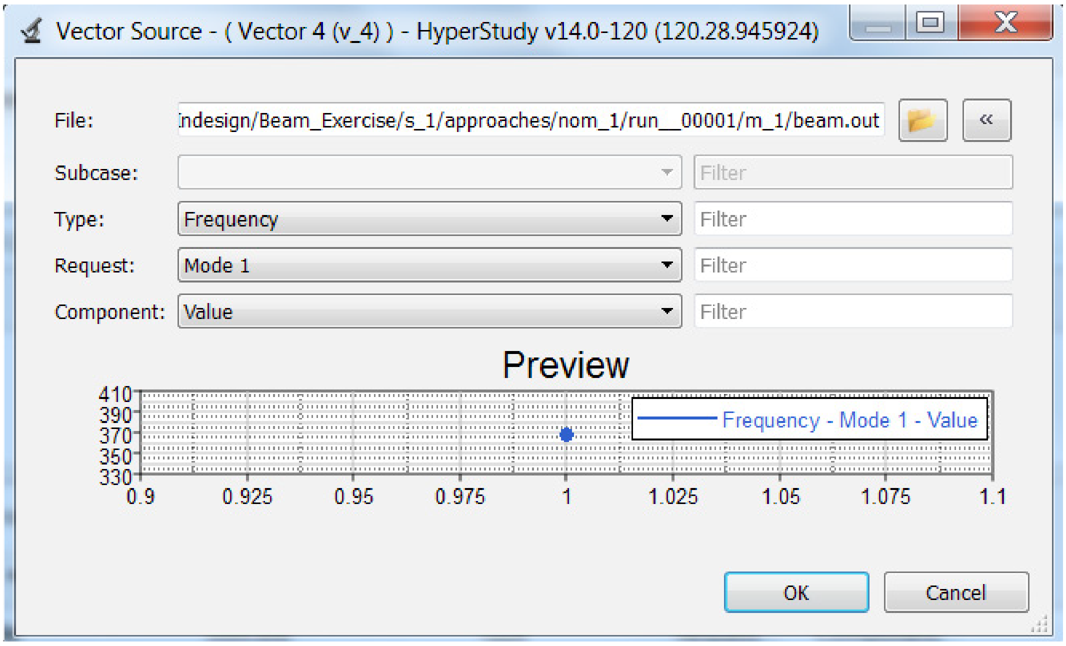


And finally, we define the response (vector) regarding the frequency – first eigenmode.

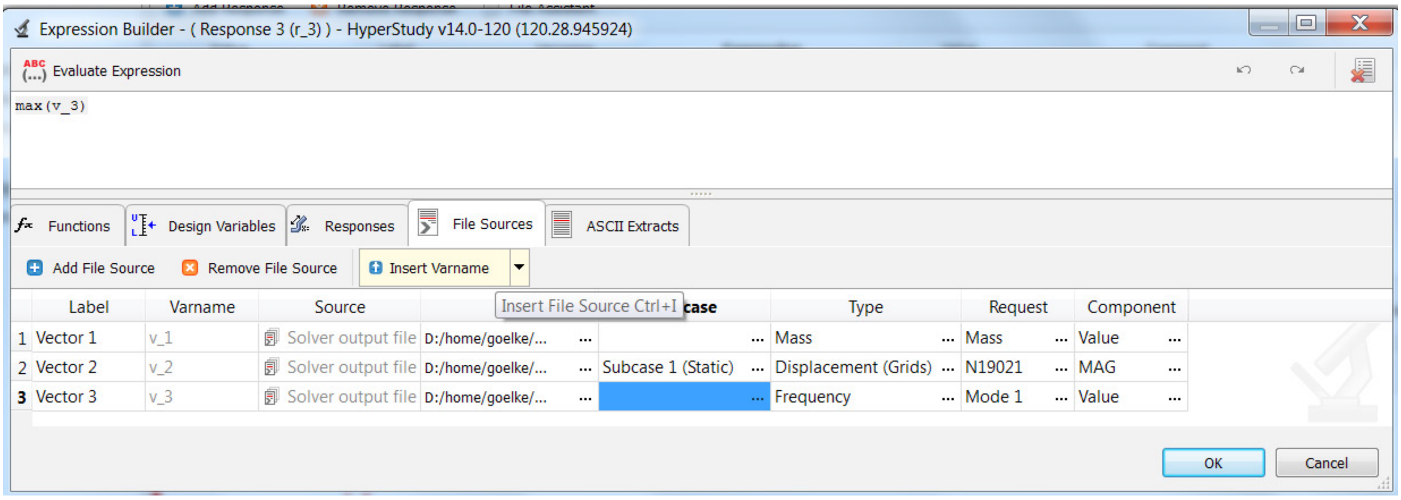
All you need to know (OptiStruct specific) is that the respective information about the eigenmode is contained in the beam.out file.

Hence, knowing this, the process to create the response of type eigenfrequency is just like before:

- Add new Response
- Activate “Expression …”
- Add File Source→ Type: Solver output file
- In the Vector Source panel, specify “File …” to be beam.out, Type: Frequency, Request Mode 1



Then activate: "Insert Varname" to show the vector in the Expression Builder Evaluate Expression (should then prompt the value 368.8311)



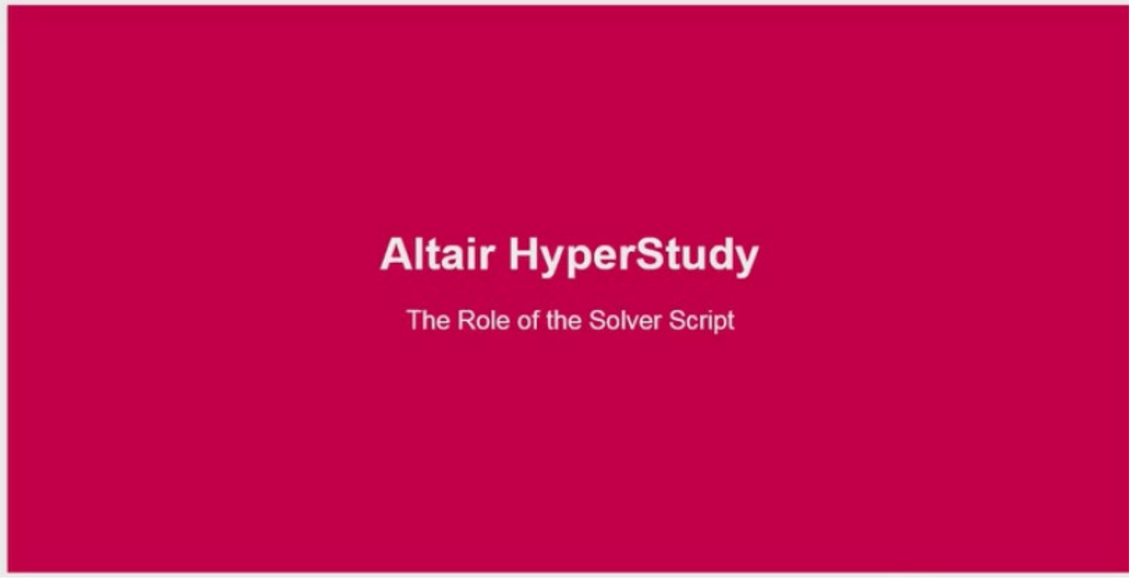
8.4 The Role of the Solver Script

A HyperStudy model is a construct that maps a set of independent variables to a set of dependent responses. The model consists of three separate steps: writing, executing, and extracting. The writing of the model files is the process of getting the independent variable out of HyperStudy and into another format. Most of the time this corresponds to the writing of a file in some format. The opposite is true in the extraction step: data exists in some output files and must be absorbed into HyperStudy. The solver script is responsible for bridging the gap in moving the process along from the input file to the output files. So the question to ask is: "Given an input file, how do I generate the output files?".

Knowing how to generate the output files will provide the directions on how to construct the solver script. Imagine being given the input file and asked to get the results file. If the process is as simple as submitting the file as direct input to some program's executable, then the solver script could be as simple as a single line. The solver script should not finish and return control back to HyperStudy until both the process is completed and the output files are generated. If there are many steps involved, such as uploading the file to a server, submitting the file to a queue, waiting for it to finish, and then downloading the file, then the solver script must perform all these actions, too.

A solver script must be able to do everything you would do manually to create the outputs given the input file, but non-interactively.

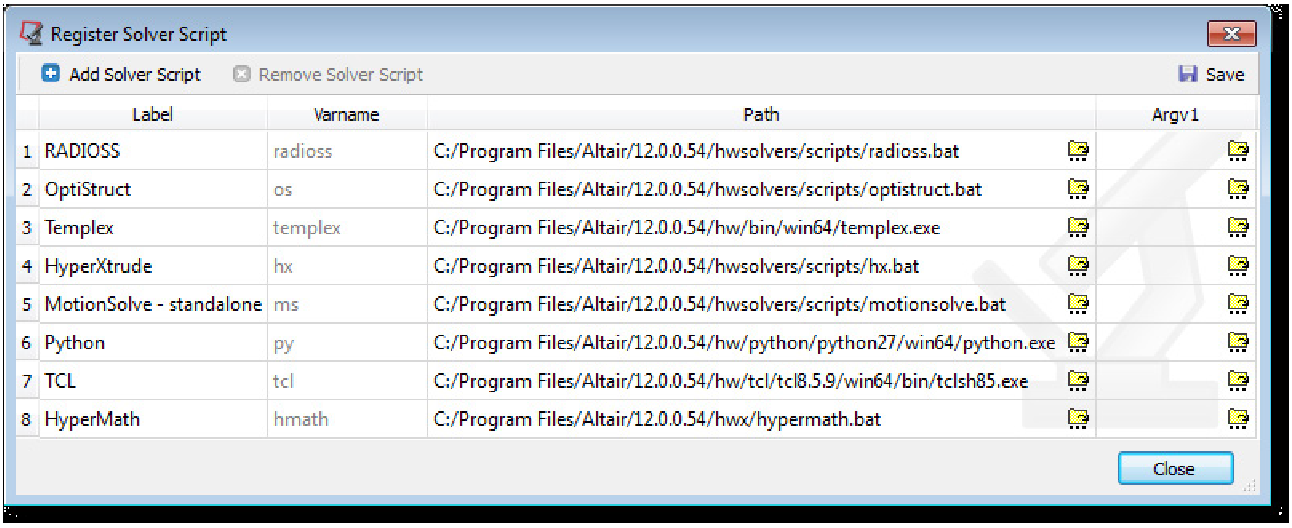
Our colleague Joseph Pajot summarized some of these aspects in the following video:



https://altair-2.wistia.com/medias/0036d3cbek

In HyperStudy, each model is associated with a solver execution script. The role of the solver script is to provide the name and location of the file that HyperStudy uses to execute the model. There is no unique method to the construction of these files. The scripts can be constructed in any language, and its contents can be as simple as a single line or a detailed set of commands. This generality is intentional so that HyperStudy remains flexible enough to be wrapped around any non-interactive process.

For more information on how to register solver script files in HyperStudy, refer to Register Solver Scripts (see Online Help Documentation HyperStudy → More on Files)



	Label	Varname	Path	Argv 1
1	RADIOSS	radioss	C:/Program Files/Altair/12.0.0.54/hwsolvers/scripts/radioss.bat	
2	OptiStruct	os	C:/Program Files/Altair/12.0.0.54/hwsolvers/scripts/optistruct.bat	
3	Templex	templex	C:/Program Files/Altair/12.0.0.54/hw/bin/win64/templex.exe	
4	HyperXtrude	hx	C:/Program Files/Altair/12.0.0.54/hwsolvers/scripts/hx.bat	
5	MotionSolve - standalone	ms	C:/Program Files/Altair/12.0.0.54/hwsolvers/scripts/motionsolve.bat	
6	Python	py	C:/Program Files/Altair/12.0.0.54/hw/python/python27/win64/python.exe	
7	TCL	tcl	C:/Program Files/Altair/12.0.0.54/hw/tcl/tcl8.5.9/win64/bin/tclsh85.exe	
8	HyperMath	hmath	C:/Program Files/Altair/12.0.0.54/hwx/hypermath.bat	

For each run, HyperStudy creates a separate run folder. In the case of multiple models, a separate model folder is also created.

These folders are called Study Run folders.

For each run, HyperStudy writes the solver input file to the Study Run folder.

If any other files need to reside in the Study Run folder, they will need to be copied. For example, RADIOSS needs a starter file and an engine file to reside in the Study Run folder. A file can be copied from the study directory by adding its name to the solver input file. Separate names in the solver input file with a semi-colon

In order for HyperStudy to execute properly, verify that the solver returns control back to HyperStudy only after the execution is finished. Otherwise, HyperStudy will attempt to extract results before all files are finished writing and the study will fail. In order to avoid, this, if possible, the solver should be run in interactive mode. Otherwise, you will need to include a wait command in your batch file.

For Abaqus, this is <PATH>/abaqus.exe job=jobname.inp interactive

In a study that uses more than one model, the models are executed in a sequence determined by HyperStudy. To control the sequence of runs, specify the priority option for the model. The results are extracted after the solvers have finished, or earlier depending on any model dependencies.

A failure during the script execution can be noted by creating a file titled task__exe_err.txt. If this file is present in the run directory, HyperStudy will detect the execution as a failure. A similar error file can be created for other task failures: write (task__wri_err.txt), extraction (task__ext_err.txt) and purge (task__pur_err.txt).

8.5 Related Material Found on the Web

We found the below listed references quite interesting and helpful. Please note that there is no business relationship (association) between the authors and Altair – the list below is included for your convenience only and is by far not complete.

If you find other material more helpful, please let us know - we would be happy to include your favorite references in here as well.

ENGINEERING STATISTICS HANDBOOK

HOME TOOLS & AIDS SEARCH BACK NEXT

5. Process Improvement
5.3. Choosing an experimental design
5.3.3. How do you select an experimental design?
5.3.6. Response surface designs

5.3.3.6.2. Box-Behnken designs

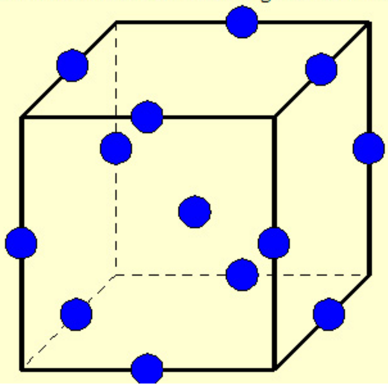
An alternate choice for fitting quadratic models that requires 3 levels of each factor and is rotatable (or "nearly" rotatable)

The Box-Behnken design is an independent quadratic design in that it does not contain an embedded factorial or fractional factorial design. In this design the treatment combinations are at the midpoints of edges of the process space and at the center. These designs are rotatable (or near rotatable) and require 3 levels of each factor. The designs have limited capability for orthogonal blocking compared to the central composite designs.

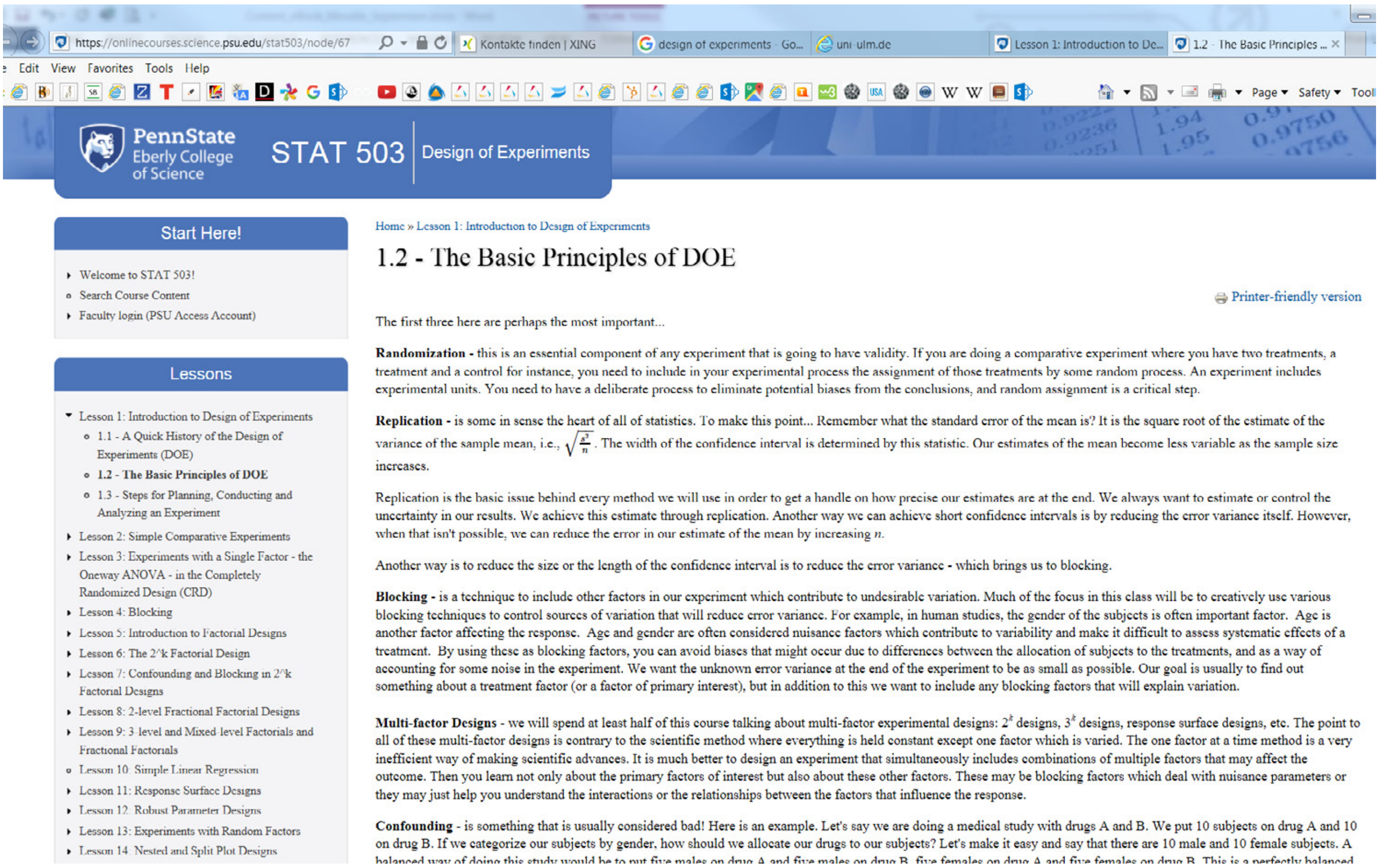
Figure 3.22 illustrates a Box-Behnken design for three factors.

Box-Behnken design for 3 factors

FIGURE 3.22 A Box-Behnken Design for Three Factors



http://itl.nist.gov/div898/handbook/pri/section3/pri3362.htm



Home » Lesson 1: Introduction to Design of Experiments

1.2 - The Basic Principles of DOE

Printer-friendly version

The first three here are perhaps the most important...

Randomization - this is an essential component of any experiment that is going to have validity. If you are doing a comparative experiment where you have two treatments, a treatment and a control for instance, you need to include in your experimental process the assignment of those treatments by some random process. An experiment includes experimental units. You need to have a deliberate process to eliminate potential biases from the conclusions, and random assignment is a critical step.

Replication - is some in sense the heart of all of statistics. To make this point... Remember what the standard error of the mean is? It is the square root of the estimate of the variance of the sample mean, i.e., $\sqrt{\frac{s^2}{n}}$. The width of the confidence interval is determined by this statistic. Our estimates of the mean become less variable as the sample size increases.

Replication is the basic issue behind every method we will use in order to get a handle on how precise our estimates are at the end. We always want to estimate or control the uncertainty in our results. We achieve this estimate through replication. Another way we can achieve short confidence intervals is by reducing the error variance itself. However, when that isn't possible, we can reduce the error in our estimate of the mean by increasing n.

Another way is to reduce the size or the length of the confidence interval is to reduce the error variance - which brings us to blocking.

Blocking - is a technique to include other factors in our experiment which contribute to undesirable variation. Much of the focus in this class will be to creatively use various blocking techniques to control sources of variation that will reduce error variance. For example, in human studies, the gender of the subjects is often important factor. Age is another factor affecting the response. Age and gender are often considered nuisance factors which contribute to variability and make it difficult to assess systematic effects of a treatment. By using these as blocking factors, you can avoid biases that might occur due to differences between the allocation of subjects to the treatments, and as a way of accounting for some noise in the experiment. We want the unknown error variance at the end of the experiment to be as small as possible. Our goal is usually to find out something about a treatment factor (or a factor of primary interest), but in addition to this we want to include any blocking factors that will explain variation.

Multi-factor Designs - we will spend at least half of this course talking about multi-factor experimental designs: 2^k designs, 3^k designs, response surface designs, etc. The point to all of these multi-factor designs is contrary to the scientific method where everything is held constant except one factor which is varied. The one factor at a time method is a very inefficient way of making scientific advances. It is much better to design an experiment that simultaneously includes combinations of multiple factors that may affect the outcome. Then you learn not only about the primary factors of interest but also about these other factors. These may be blocking factors which deal with nuisance parameters or they may just help you understand the interactions or the relationships between the factors that influence the response.

Confounding - is something that is usually considered bad! Here is an example. Let's say we are doing a medical study with drugs A and B. We put 10 subjects on drug A and 10 on drug B. If we categorize our subjects by gender, how should we allocate our drugs to our subjects? Let's make it easy and say that there are 10 male and 10 female subjects. A

https://onlinecourses.science.psu.edu/stat503/node/67

As you may can imagine, the list of videos about DOE found in the internet, especially on YouTube is endless.

Just search for: Design of Experiments, Intro to DOE, Intractions & Main Effects, Interpreting Main Effects, DOE Examples and so on.

Please let us know your favorite DOE related videos - we are happy to include them in here.