Workshop - Automated Optimization of a Composite Laminate

AN MSC NASTRAN SOL 200 TUTORIAL

Questions? Email: christian@ the-engineering-lab.com

Goal: Use Nastran SOL 200 Optimization

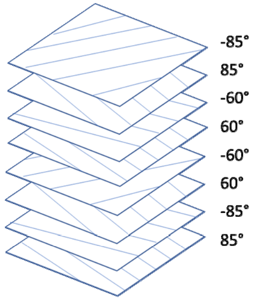
Objective: Minimize the weight of this cylinder composed of a composite laminate

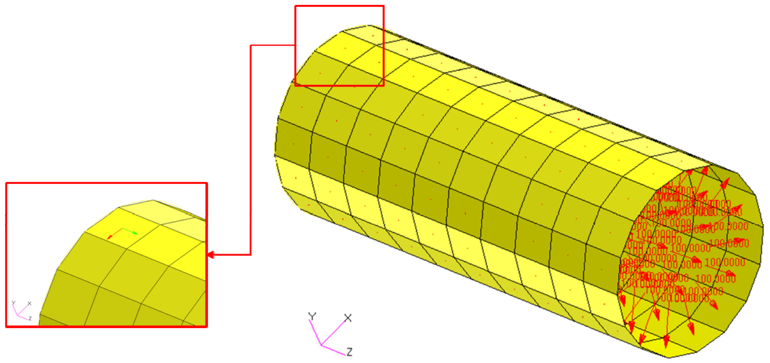
Before Optimization

- Weight: 1.60 $lb_f \cdot s^2/in$
- Layup: [85/-85/60/-60/60/-60/85/-85]
- Thickness: .0100 in
- Plies are initially in failure

After Optimization

- Weight: 1.04 $lb_f \cdot s^2/in$
- Layup: [90/-90/0/0/0/0/90/-90]
- Thickness: .0065 in

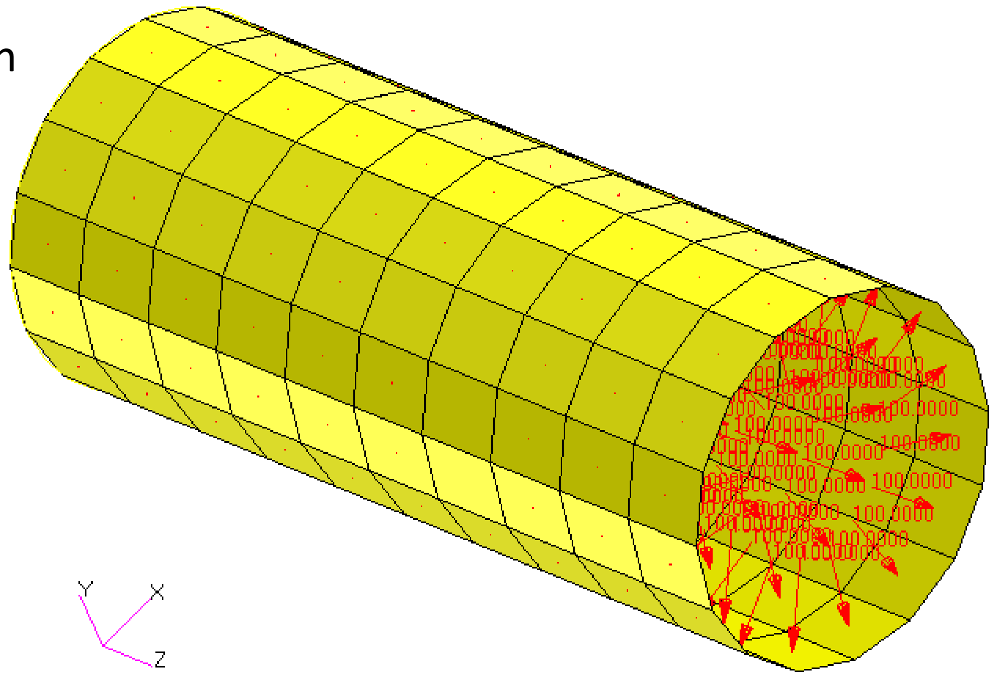




Details of the Structural Model

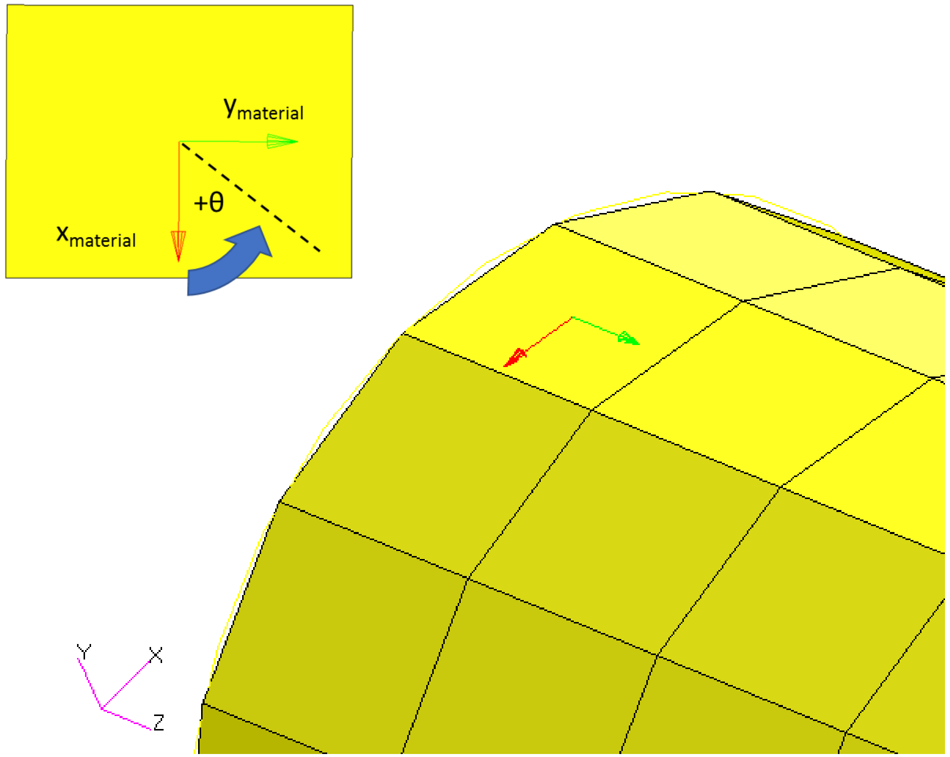
Allowed to translate in x and y

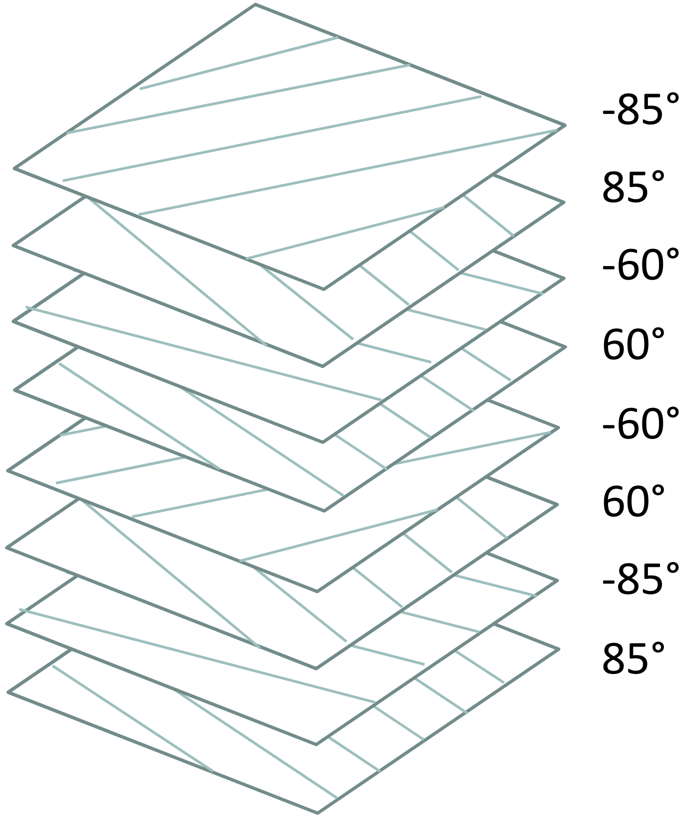
Translation prevented in z direction



Internal pressure load of 400 psi

Details of the Structural Model





Optimization Problem Statement

Design Variables

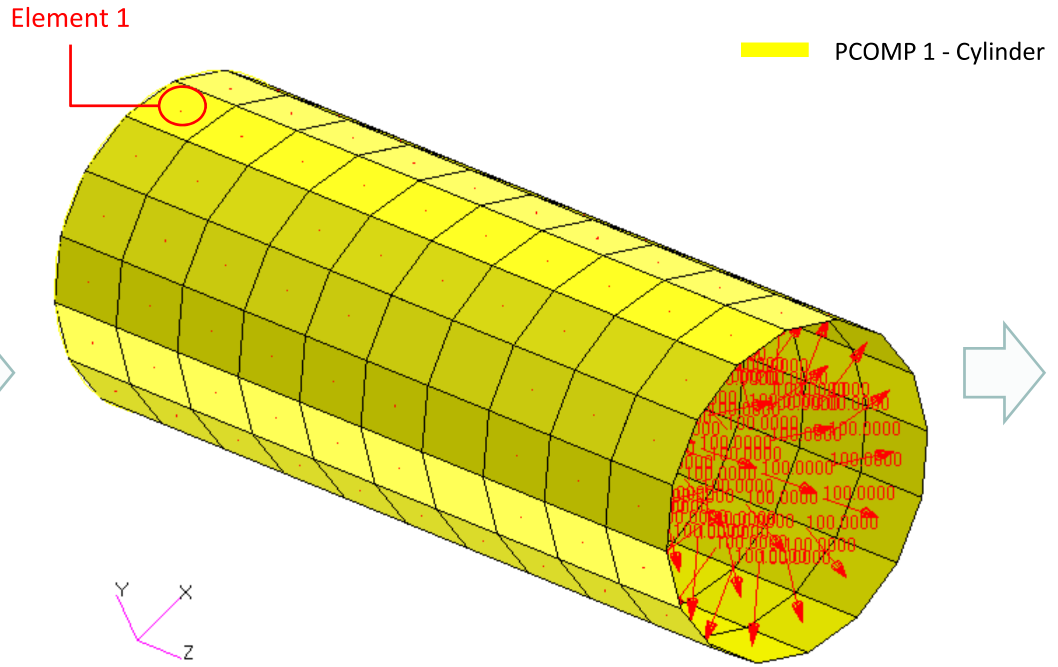
x1: T of lamina 1 of PCOMP 1
x2: T of lamina 2
x3: T of lamina 3
x4: T of lamina 4
x5: T of lamina 5
x6: T of lamina 6
x7: T of lamina 7
x8: T of lamina 8
.001 < xi
x9: Orientation of lamina 1 of PCOMP 1
x10: Orientation of lamina 2
x11: Orientation of lamina 3
x12: Orientation of lamina 4
x13: Orientation of lamina 5
x14: Orientation of lamina 6
x15: Orientation of lamina 7
x16: Orientation of lamina 8

y1: Thickness of layers
y2: Angle of layers (85)
y3: Angle of layers (-85)

.001 < y1 < 10.
-90. < y2 < 90.
-90. < y3 < 90.

Variable Linking
X1, x2, x3, … , x8 = y1
x9, x15 = y2
x10, x16= -1.0 * y2
x11 , x13 = y3
x12, x14 = -1.0 * y3

Allowed Discrete Values
y2, y3: -90, -85, …, 85, 90



Design Objective

r0: Minimize weight

Design Constraints

r1: Failure index of lamina 1 of element 1
…
r8: Failure index of lamina 8 of element 1

r1, … r8 < .9

Design Variables

Thickness Variables

Layer 8: x8 = y1

Layer 7: x7 = y1

Layer 6: x6 = y1

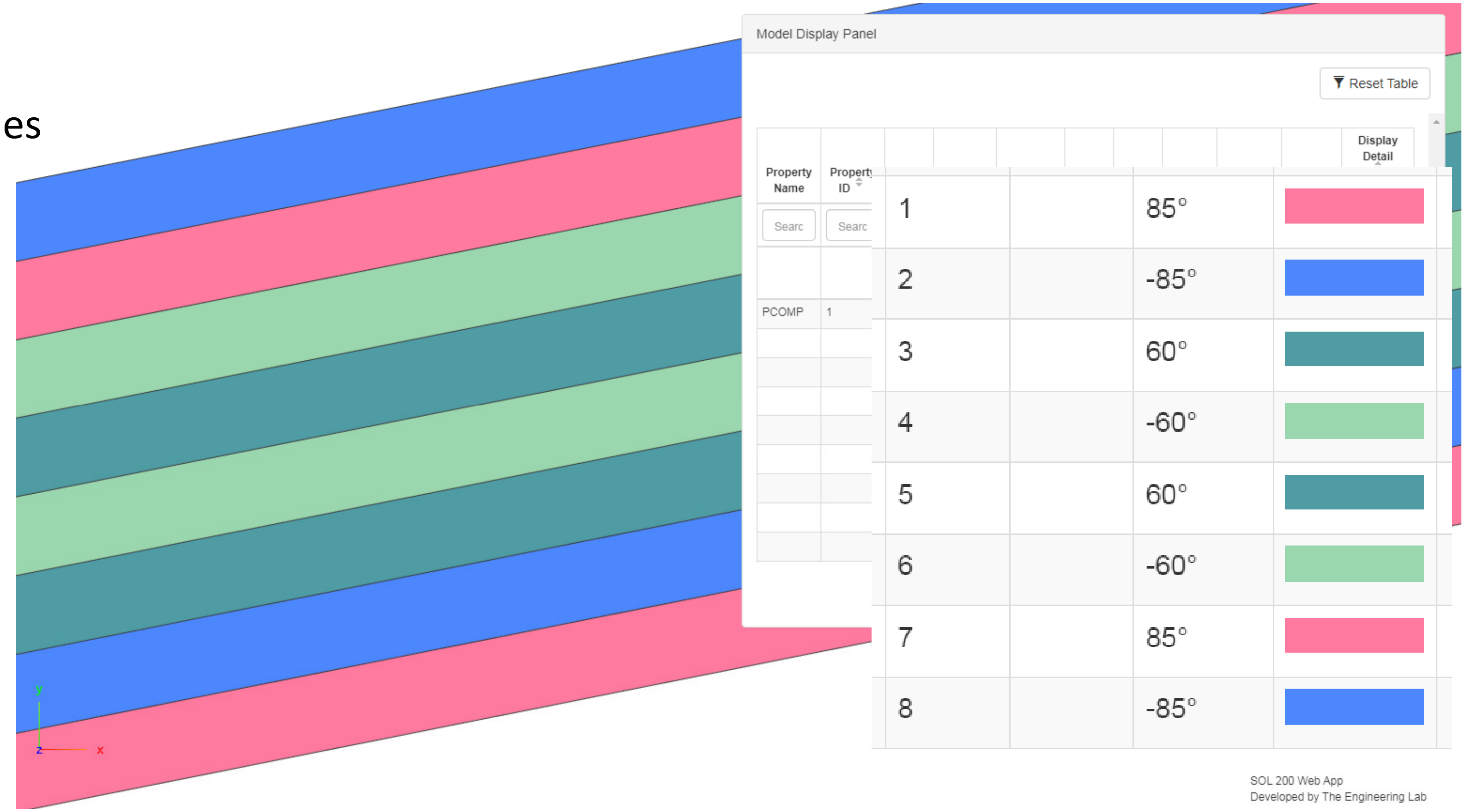
Layer 5: x5 = y1

Layer 4: x4 = y1

Layer 3: x3 = y1

Layer 2: x2 = y1

Layer 1: x1 = y1



Design Variables

Angle Variables

Layer 8: x16 = -y2

Layer 7: x15 = y2

Layer 6: x14 = -y3

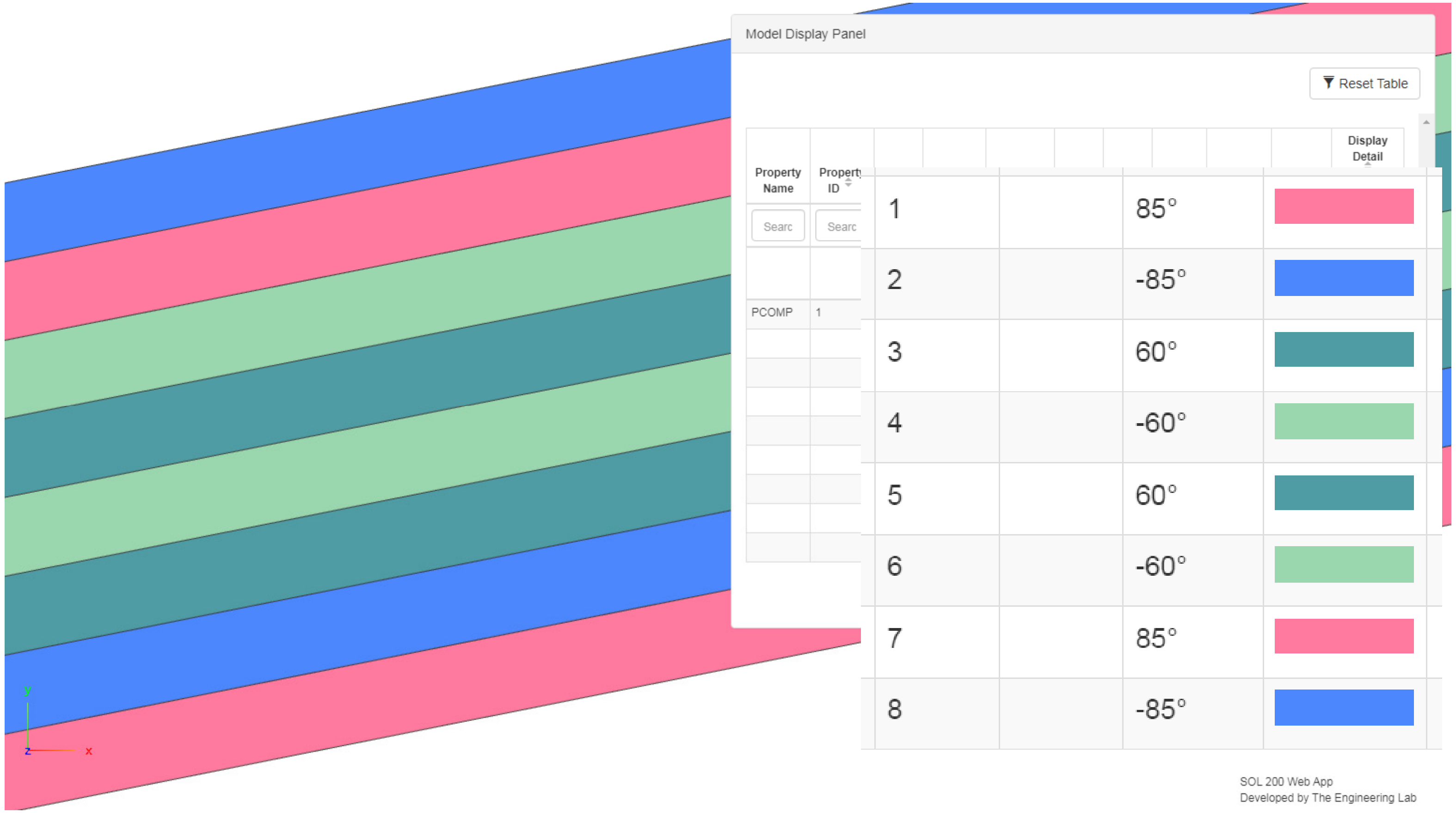
Layer 5: x13 = y3

Layer 4: x12 = -y3

Layer 3: x11 = y3

Layer 2: x10 = -y2

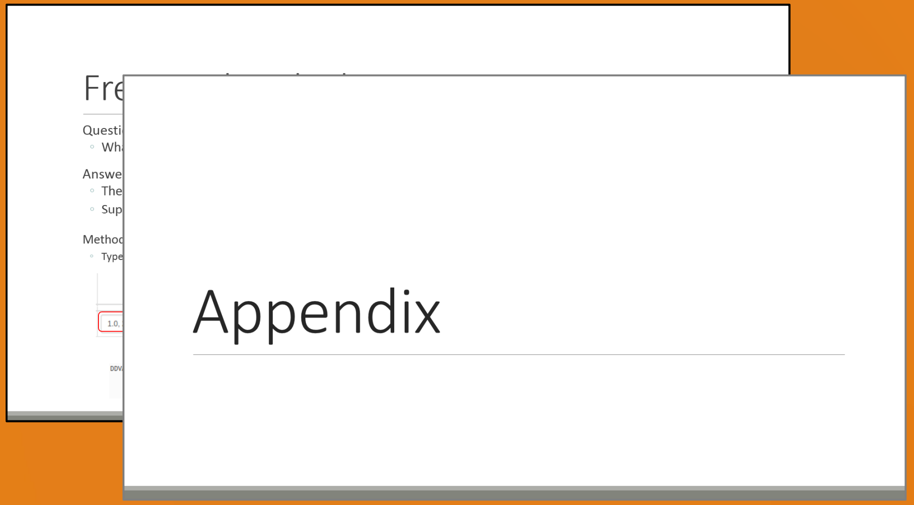
Layer 1: x9 = y2



More Information Available in the Appendix

The Appendix includes information regarding the following:

- Frequently Asked Questions
  - What are the ways of specifying allowable discrete values for design variables?
  - Why do the orientation angles not change?



Contact me

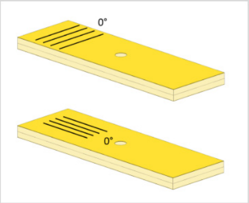
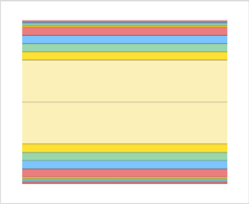
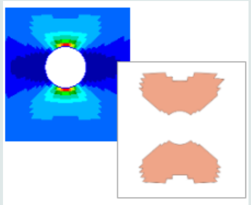
- Nastran SOL 200 training
- Nastran SOL 200 questions
- Structural or mechanical optimization questions
- Access to the SOL 200 Web App

christian@ the-engineering-lab.com

Before Continuing

Consider the New Composite Laminate Optimization Tutorials – Composite Coupon

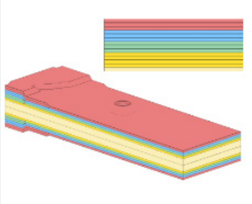
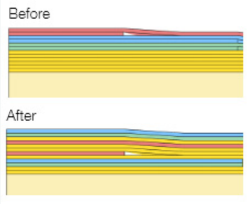
Visit the User's Guide to access the newest tutorials.

	Title and Description	PDF Tutorial	YouTube Tutorial
	**Composite Coupon – Phase A – Determination of the optimal 0° direction of a composite** The goal of this 5-phase tutorial series is to optimize a composite coupon, with a core, and produce a lightweight composite that satisfies failure index constraints. The optimal ply shapes (ply drop-offs) and ply numbers are determined for 0°, ±45°, and 90° plies. A stacking sequence optimization is performed to satisfy manufacturing requirements. One important part of optimizing composites is visualizing the composite plies. This tutorial series also demonstrates the visualization of ply drop-offs, tapered plies and core layers. This first phase involves determining the optimal 0° direction of a composite. It is best practice to align the 0° plies in the direction of the load. Not doing so will more than likely produce a suboptimal composite that is heavier than necessary. This tutorial demonstrates the use of MSC Nastran's optimizer to determine the optimal 0° direction of a composite. An optimization is performed to maximize the stiffness of the composite for multiple load cases and while varying the angle of the 0° plies. Ultimately, the best 0° direction is determined. This is the first phase in a 5-phase tutorial series.	Link	Link
	**Composite Coupon – Phase B – Baseline Ply Number Optimization** This tutorial demonstrates how to configure a basic ply number optimization of continuous plies that span the entire model. The goal of this tutorial is to demonstrate basic actions such as creating variables, a weight objective and constraints on failure index. The results of this ply number optimization serve as a baseline for future comparisons. In a subsequent tutorial, the ply shapes will be optimized to minimize weight. This is the second phase in a 5-phase tutorial series.	Link	Link
	**Composite Coupon – Phase C – Data Preparation for Ply Shape Optimization** This tutorial is a guide to preparing data for ply shape optimization in a subsequent tutorial. The maximum failure index values of the outer plies of the composite are determined and saved to specially formatted PLY000i files. The PLY000i files will be used to construct optimal ply shapes in a subsequent tutorial. This is the third phase in a 5-phase tutorial series.	Link	Link

Before Continuing

Consider the New Composite Laminate Optimization Tutorials – Composite Coupon, Continued

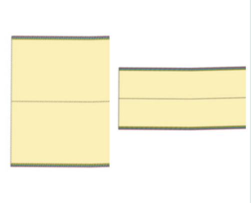
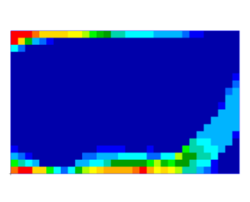
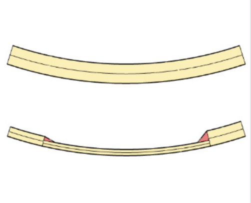
Visit the User’s Guide to access the newest tutorials.

	Title and Description	PDF Tutorial	YouTube Tutorial
	**Composite Coupon – Phase D – Ply Shape and Ply Number Optimization** This tutorial details the process to build optimal ply shapes and perform a ply number optimization. The optimal ply shapes are constructed to follow the contours of the failure indices. The ply number optimization involves minimizing weight and constraining the failure indices of plies. The PLY000i files and BDF files from the previous tutorial, phase C, are used in this tutorial. This is the fourth phase in a 5-phase tutorial series.	Link	Link
	**Composite Coupon – Phase E – Stacking Sequence Optimization** This tutorial involves performing a stacking sequence optimization and is a continuation of the previous tutorial, phase D. A final statics analysis is performed to confirm the optimized composite satisfies failure index constraints. This is the fifth phase in a 5-phase tutorial series.	Link	Link

Before Continuing

Consider the New Composite Laminate Optimization Tutorials – Sandwich Composite Panel

Visit the User's Guide to access the newest tutorials.

	Title and Description	PDF Tutorial	YouTube Tutorial
	**Sandwich Composite Panel – Phase B – Baseline Core Thickness Optimization** The goal of this 3-phase tutorial series is to optimize a curved composite panel, with a core, and produce a lightweight composite that satisfies constraints on the buckling load factor. This tutorial series focuses exclusively on optimizing the thickness of the core. The methods detailed in the tutorial series are applicable to both foam and honeycomb cores. This tutorial demonstrates how to configure a basic core thickness optimization where the core has a constant thickness throughout the entire model. The goal of this tutorial is to demonstrate basic actions such as creating variables, a weight objective and constraints on the buckling load factor. The results of this core thickness optimization serve as a baseline for future comparisons. In a subsequent tutorial, the core will be allowed to have a variable thickness throughout the model and will be optimized to minimize weight. This is the first phase in a 3-phase tutorial series.	Link	Link
	**Sandwich Composite Panel – Phase C – Topometry Optimization to Determine Optimal Core Shape** This tutorial is a guide to preparing data for core shape and core thickness optimization in a subsequent tutorial. A topometry optimization is performed in this tutorial to determine the ideal thickness distribution of the core throughout the entire composite panel while satisfying constraints on the buckling load factor and minimizing weight. The results of a topometry optimization are contained in the PLY000i files and will be used to construct optimal core shapes in a subsequent tutorial. This is the second phase in a 3-phase tutorial series.	Link	Link
	**Sandwich Composite Panel – Phase D – Core Shape and Core Thickness Optimization** This tutorial details the process to build optimal core shapes and perform a core thickness optimization. The optimal core shapes are constructed to follow the contours of thickness results generated by a topometry optimization. The core thickness optimization involves minimizing weight and constraining the buckling load factor. The PLY000i files and BDF files from the previous tutorial, phase C, are used in this tutorial. Comparisons are made between this optimization in phase D and the baseline optimization performed in phase B. This is the third phase in a 3-phase tutorial series.	Link	Link

Tutorial

Tutorial Overview

1. Start with a .bdf or .dat file
2. Use the SOL 200 Web App to:
   - Convert the .bdf file to SOL 200
     - Design Variables
     - Design Objective
     - Design Constraints
   - Perform optimization with Nastran SOL 200
3. Plot the Optimization Results
4. Update the original model with optimized parameters

Special Topics Covered

Discrete Values for Design Variables - This example has a requirement where the design variables can only take on specific values. Instead of an optimization solution where the values may be 45.23423 or 15.90234, the use of Discrete Values will allow specific values to be obtained such as 45.0 or 16.0. This tutorial showcases a feature to specify specific values that can be taken by the design variables.

Design Variable Linking - In some situations, one design variable will drive the values of other design variables. For example, the thickness of one section will be the same for other sections. This tutorial demonstrates the use of Design Variable Linking to address such examples.

$$x2 = x1$$

$$x4 = x1$$

$$x5 = x1$$

SOL 200 Web App Capabilities

The Post-processor Web App and HDF5 Explorer are free to MSC Nastran users.

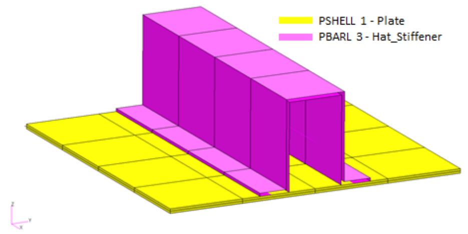
Compatibility

- Google Chrome, Mozilla Firefox or Microsoft Edge
- Windows and Red Hat Linux
- Installable on a company laptop, workstation or server. All data remains within your company.

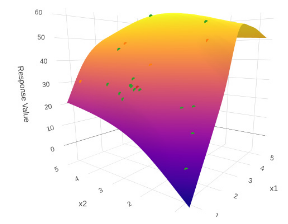
Benefits

- REAL TIME error detection. 200+ error validations.
- REALT TIME creation of bulk data entries.
- Web browser accessible
- Free Post-processor web apps
- +80 tutorials

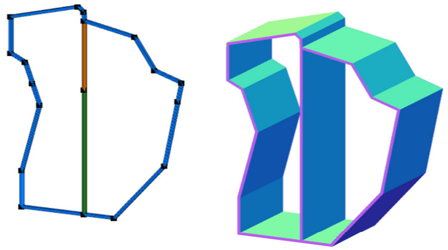
Web Apps



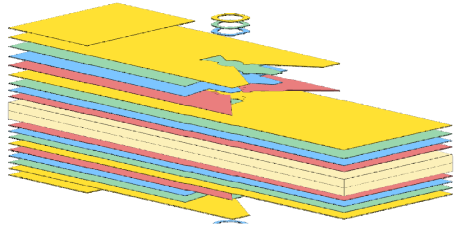
Web Apps for MSC Nastran SOL 200
Pre/post for MSC Nastran SOL 200. Support for size, topology, topometry, topography, multi-model optimization.



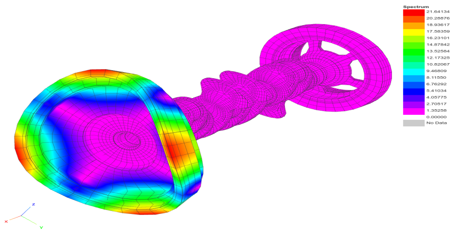
Machine Learning Web App
Bayesian Optimization for nonlinear response optimization (SOL 400)



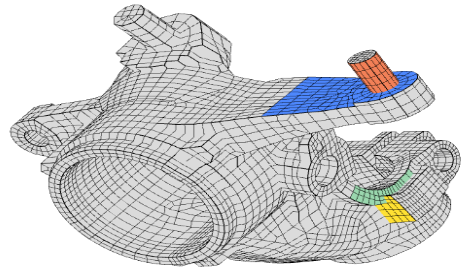
PBMSECT Web App
Generate PBMSECT and PBRSECT entries graphically



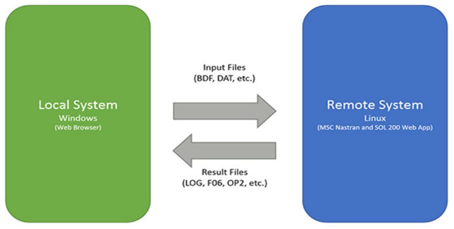
Ply Shape Optimization Web App
Optimize composite ply drop-off locations, and generate new PCOMPG entries



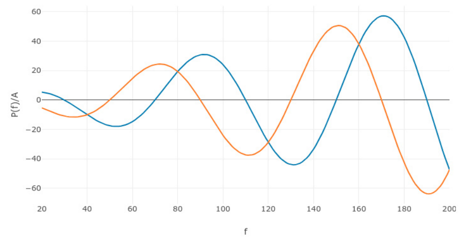
Post-processor Web App
View MSC Nastran results in a web browser on Windows and Linux



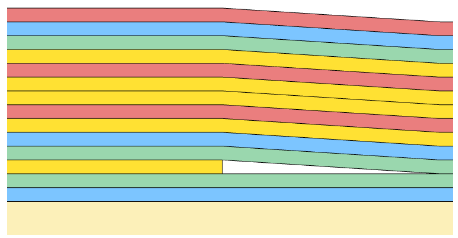
Shape Optimization Web App
Use a web application to configure and perform shape optimization.



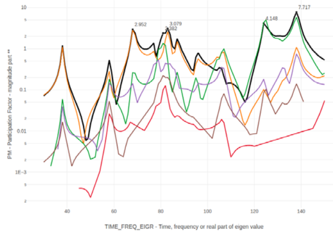
Remote Execution Web App
Run MSC Nastran jobs on remote Linux or Windows systems available on the local network



Dynamic Loads Web App
Generate RLOAD1, RLOAD2 and DLOAD entries graphically



Stacking Sequence Web App
Optimize the stacking sequence of composite laminate plies

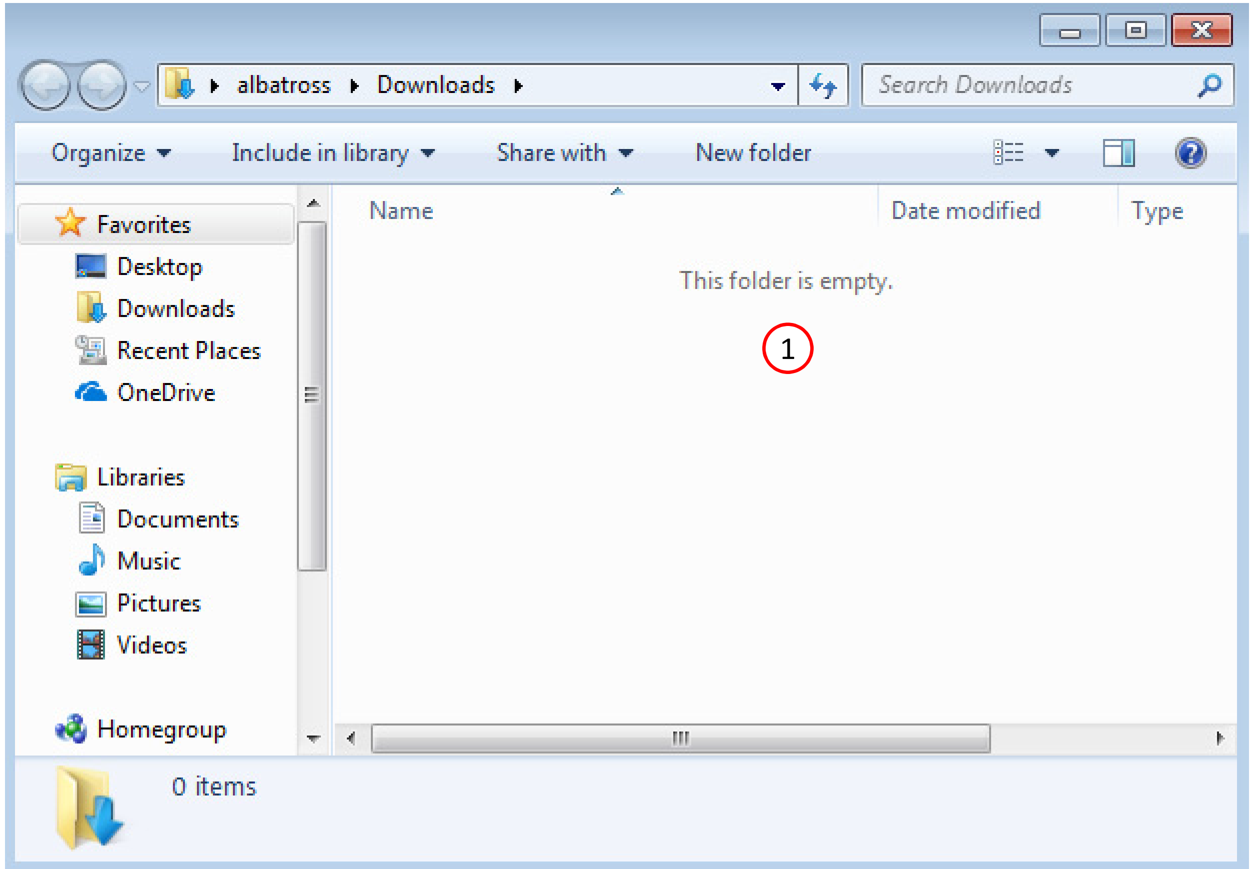


HDF5 Explorer Web App
Create graphs (XY plots) using data from the H5 file

Before Starting

1. Ensure the Downloads directory is empty in order to prevent confusion with other files

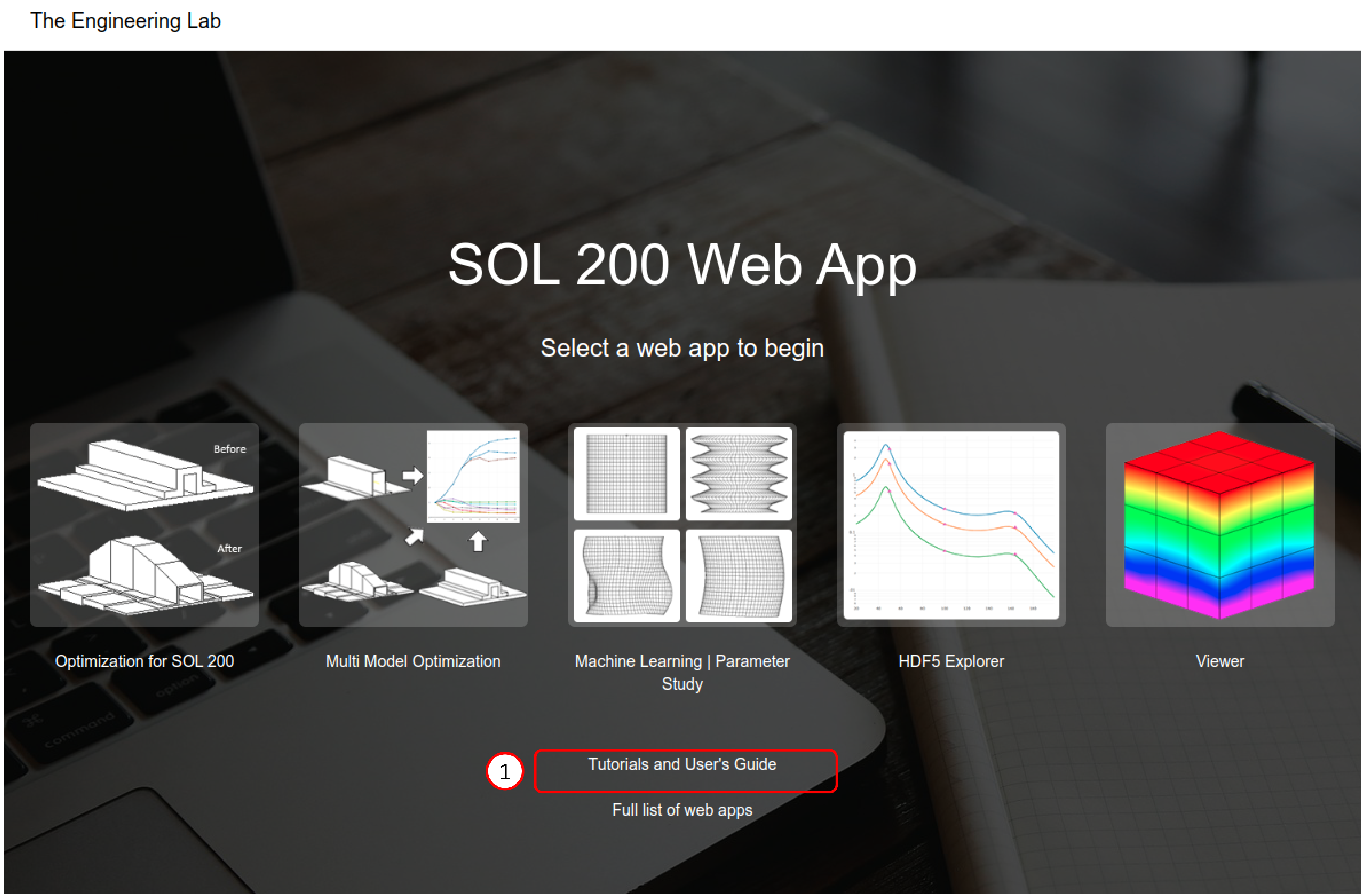
- Throughout this workshop, you will be working with multiple file types and directories such as:
    - .bdf/.dat
    - nastran_working_directory
    - .f06, .log, .pch, .h5, etc.
- To minimize confusion with files and folders, it is encouraged to start with a clean directory.



Go to the User's Guide

1. Click on the indicated link

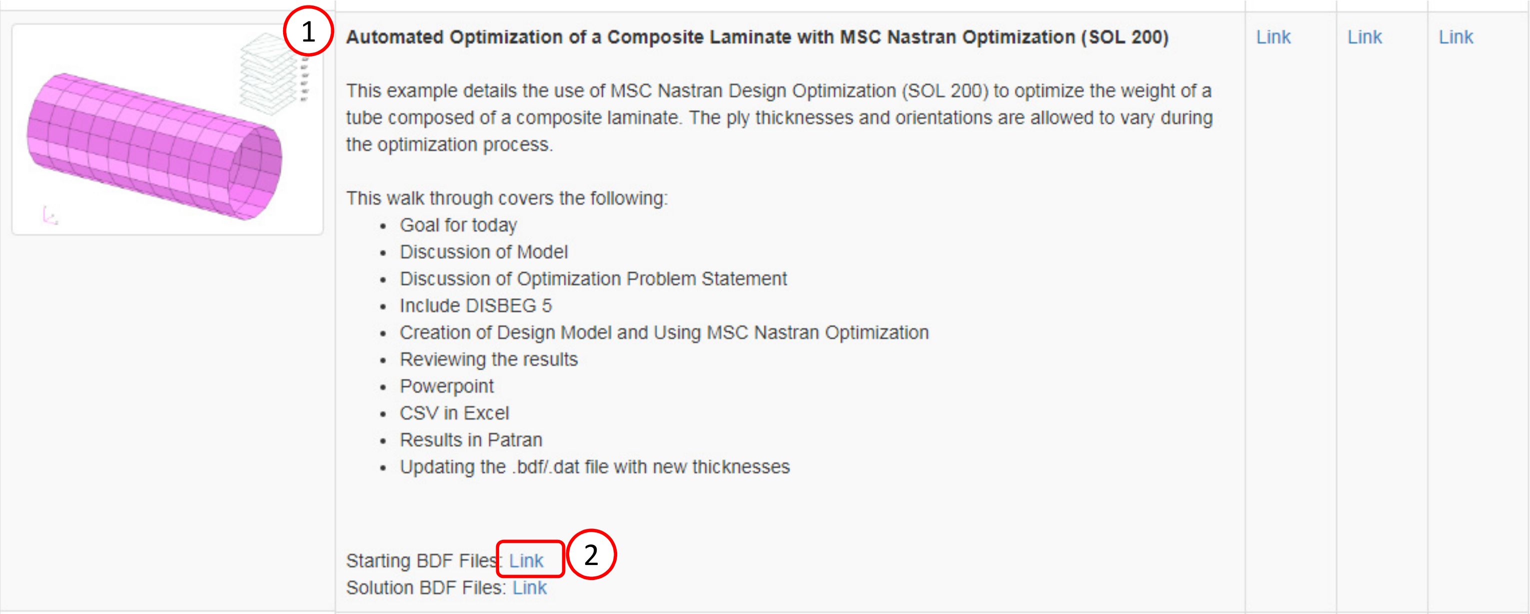
- The necessary BDF files for this tutorial are available in the Tutorials section of the User's Guide.

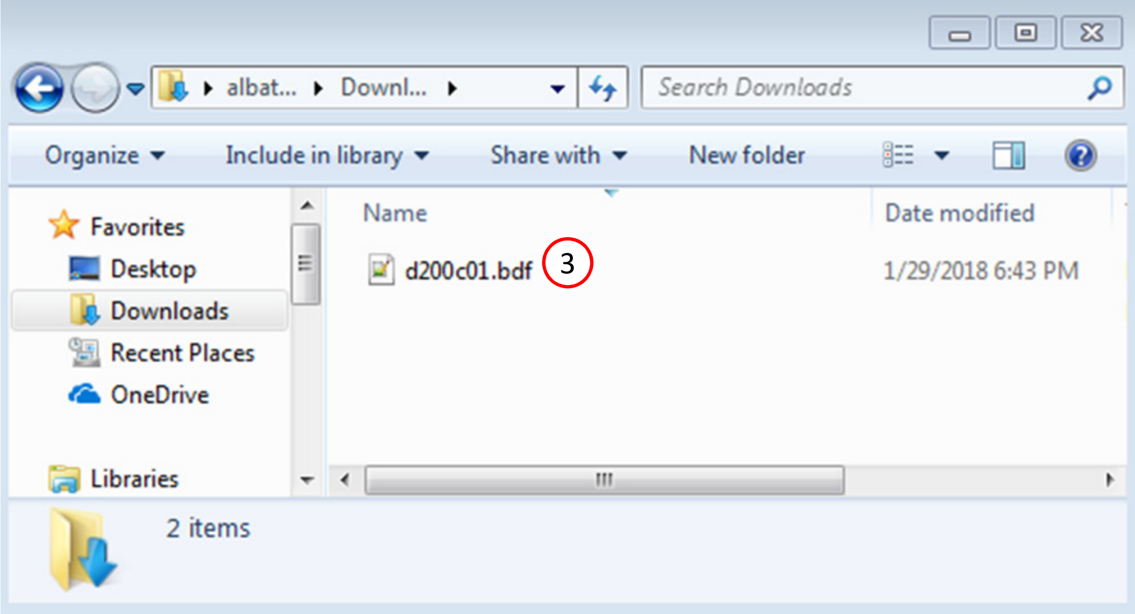


Obtain Starting Files

1. Find the indicated example
2. Click Link
3. The starting file has been downloaded

- When starting the procedure, all the necessary BDF files must be collected together.

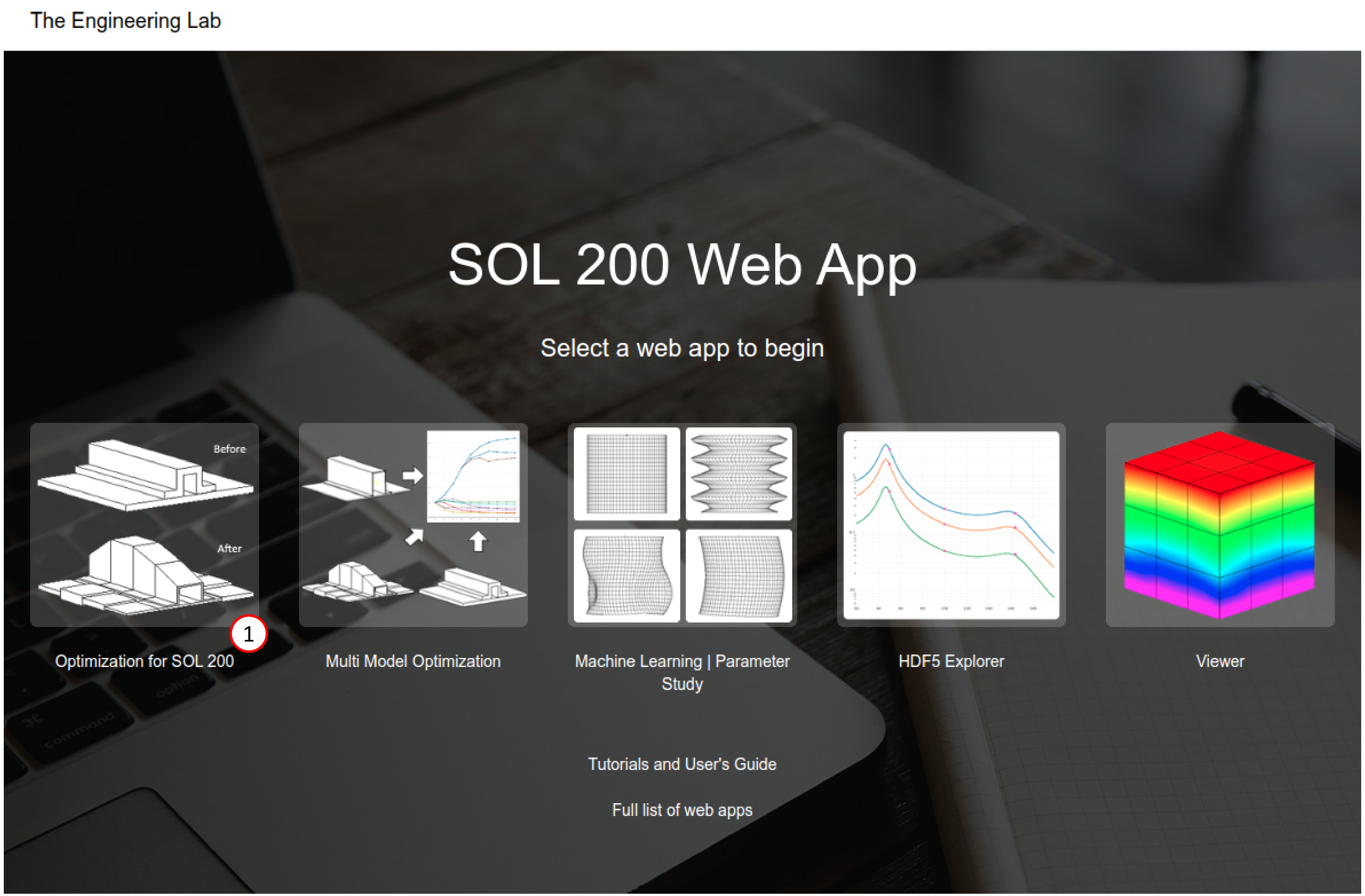




Open the Correct Page

1. Click on the indicated link

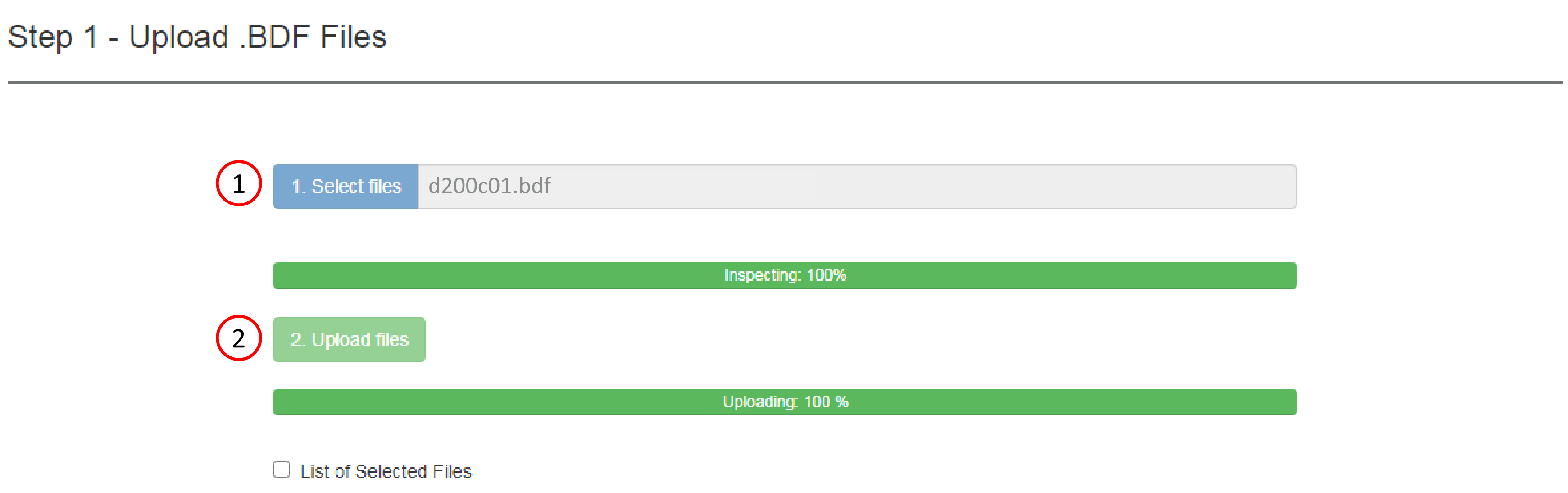
- MSC Nastran can perform many optimization types. The SOL 200 Web App includes dedicated web apps for the following:
  - Optimization for SOL 200 (Size, Topology, Topometry, Topography, Local Optimization, Sensitivity Analysis and Global Optimization)
  - Multi Model Optimization
  - Machine Learning
- The web app also features the HDF5 Explorer, a web application to extract results from the H5 file type.



Upload BDF Files

1. Click 1. Select Files and select d200c01.bdf
2. Click Upload Files

- The process starts by uploading all the necessary BDF files. The BDF files can be files of your own or files found in the Tutorials section of the User's Guide.



Create Design Variables

1. In the search box, type ‘t’
2. Click twice on the Property column header to sort the column in decreasing order
3. Type in 16 to show only 16 rows
4. Click + Options
5. Click Create

- There are 2 methods to create the 16 design variables: Click each blue plus icon, which requires 16 mouse clicks, OR click the yellow Create icon, which requires 1 mouse click.
- Each step has hidden functionality for advanced users. The visibility is controlled by clicking + Options.
- If the property entry, e.g. PSHELL, was given a name in Patran, e.g. Car Door, the name can be shown by marking the checkbox titled Entry Name.

Step 1 - Select design properties

+ Options

Display Type	% Lower Bound	% Upper Bound	Lower Bound	Upper Bound	Allowed Discrete Values or Equation	Bulk Create
DVXREL1	Lower	Upper	Lower	Upper	Allowed discrete values, example: -2.0, 1.0, THRU, 10.0, BY, 1.0	Create
DVXREL1 Unity	Lower	Upper	Lower	Upper	Allowed discrete values, example: -2.0, 1.0, THRU, 10.0, BY, 1.0	Create
DVXREL2	Lower	Upper	Lower	Upper	Type equation here, example: y1**2 + x2 + k3	Create

Display Columns

Create DVXREL1 Create Unity DVXREL1 Create DVXREL2 Entry Name

Settings for row filtering in tables

Contains Starts with Ends with

Create DVXREL1	Property	Property Description	Entry	Entry ID	Current Value
	t	Search	Search	Search	Search
+	T1	Thickness of ply	PCOMP	1	.01
+	T2	Thickness of ply	PCOMP	1	.01
+	T3	Thickness of ply	PCOMP	1	.01
+	T4	Thickness of ply	PCOMP	1	.01
+	T5	Thickness of ply	PCOMP	1	.01
+	T6	Thickness of ply	PCOMP	1	.01
+	T7	Thickness of ply	PCOMP	1	.01
+	T8	Thickness of ply	PCOMP	1	.01
+	THETA1	Orientation angle of ply	PCOMP	1	85.
+	THETA2	Orientation angle of ply	PCOMP	1	-85.
+	THETA3	Orientation angle of ply	PCOMP	1	60.
+	THETA4	Orientation angle of ply	PCOMP	1	-60.
+	THETA5	Orientation angle of ply	PCOMP	1	60.
+	THETA6	Orientation angle of ply	PCOMP	1	-60.
+	THETA7	Orientation angle of ply	PCOMP	1	85.
+	THETA8	Orientation angle of ply	PCOMP	1	-85.

« 1 2 »

5 10 20 30 40 50

Number of Visible Rows 16

Create Design Variables

1. Click 20 on the pagination bar
2. 16 design variables have been created
   - 8 design variables have been created for the layer thicknesses
   - 8 design variables have been created for the layer angles

- In some instances, the optimizer will vary a positive design variable and make it negative, e.g. a thickness of .08 becomes -.01 in a weight minimization optimization. Certain properties, such as thickness or beam cross sections should never be negative. The lower bound in this example is set to .001 to avoid a negative variable during the optimization.
- In some scenarios, the use of 0 degrees for the initial value of an orientation angle may be unsuccessful. An alternative is to use 360 degrees as the initial value.

Step 2 - Adjust design variables

✖ Delete Visible Rows

+ Options

2

	Label	Status	Property	Property Description	Entry	Entry ID	Initial Value	Lower Bound	Upper Bound	Allowed Discrete Values
	Search	Search	Search	Search	Search	Search	Search	Search	Search	Search
✖	x1	✔	T1	Thickness of layer 1 (85°)	PCOMP	1	.01	.001	Upper	Examples: -2.0, 1.0, THRU, 10.0,
✖	x2	✔	T2	Thickness of layer 2 (-85°)	PCOMP	1	.01	.001	Upper	Examples: -2.0, 1.0, THRU, 10.0,
✖	x3	✔	T3	Thickness of layer 3 (60°)	PCOMP	1	.01	.001	Upper	Examples: -2.0, 1.0, THRU, 10.0,
✖	x4	✔	T4	Thickness of layer 4 (-60°)	PCOMP	1	.01	.001	Upper	Examples: -2.0, 1.0, THRU, 10.0,
✖	x5	✔	T5	Thickness of layer 5 (60°)	PCOMP	1	.01	.001	Upper	Examples: -2.0, 1.0, THRU, 10.0,
✖	x6	✔	T6	Thickness of layer 6 (-60°)	PCOMP	1	.01	.001	Upper	Examples: -2.0, 1.0, THRU, 10.0,
✖	x7	✔	T7	Thickness of layer 7 (85°)	PCOMP	1	.01	.001	Upper	Examples: -2.0, 1.0, THRU, 10.0,
✖	x8	✔	T8	Thickness of layer 8 (-85°)	PCOMP	1	.01	.001	Upper	Examples: -2.0, 1.0, THRU, 10.0,
✖	x9	✔	THETA1	Theta of layer 1 (85°)	PCOMP	1	85.	Lower	Upper	Examples: -2.0, 1.0, THRU, 10.0,
✖	x10	✔	THETA2	Theta of layer 2 (-85°)	PCOMP	1	-85.	Lower	Upper	Examples: -2.0, 1.0, THRU, 10.0,
✖	x11	✔	THETA3	Theta of layer 3 (60°)	PCOMP	1	60.	Lower	Upper	Examples: -2.0, 1.0, THRU, 10.0,
✖	x12	✔	THETA4	Theta of layer 4 (-60°)	PCOMP	1	-60.	Lower	Upper	Examples: -2.0, 1.0, THRU, 10.0,
✖	x13	✔	THETA5	Theta of layer 5 (60°)	PCOMP	1	60.	Lower	Upper	Examples: -2.0, 1.0, THRU, 10.0,
✖	x14	✔	THETA6	Theta of layer 6 (-60°)	PCOMP	1	-60.	Lower	Upper	Examples: -2.0, 1.0, THRU, 10.0,
✖	x15	✔	THETA7	Theta of layer 7 (85°)	PCOMP	1	85.	Lower	Upper	Examples: -2.0, 1.0, THRU, 10.0,
✖	x16	✔	THETA8	Theta of layer 8 (-85°)	PCOMP	1	-85.	Lower	Upper	Examples: -2.0, 1.0, THRU, 10.0,

5 10 20 30 40 50

Create Design Variables

1. Scroll to section Step 4 - Adjust design variables
2. Click +Create Variable 3 times to create 3 design variables y1, y2 and y3
3. Set the following for y1
    - Initial Value: .01
    - Lower Bound: .001
    - Upper Bound: 10.
4. Set the following for y2
    - Initial Value: 85.
    - Lower Bound: -90.
    - Upper Bound: 90.
    - Allowed Discrete Values: -90., THRU, 90., BY, 5.0
5. Set the following for y3
    - Initial Value: 60.
    - Lower Bound: -90.
    - Upper Bound: 90.
    - Allowed Discrete Values: -90., THRU, 90., BY, 5.0

- The angle variables are to take on integer values, so discrete values of -90, -85, 80, ..., 0, 5, 10, ..., 80, 85, 90 are specified.

Step 4 - Adjust design variables 1

+ Options

2 + Create Variable

	Label	Status	Initial Value	Lower Bound	Upper Bound	Allowed Discrete Values
	Search	Search				
	y1	3	.01	.001	10.	Examples: -2.0, 1.0, THRU, 10.0, BY, 1.0
	y2	4	85.	-90.	90.	-90., THRU, 90., BY, 5.0
	y3	5	60.	-90.	90.	-90., THRU, 90., BY, 5.0

Create Design Variables

1. Click 5 times on +Create DLINK
2. Create design variables links for the thicknesses
    1. Dependent Design Variables: x1, x2, x3, x4, x5, x6, x7, x8
    2. Equation: y1 * 1.0
3. Create design variables links for the -85 degree plies
    1. Dependent Design Variables: x10, 16
    2. Equation: y2 * -1.0
4. Create design variables links for the 85 degree plies
    1. Dependent Design Variables: x9, x15
    2. Equation: y2 * 1.0
5. Create design variables links for the -60 degree plies
    1. Dependent Design Variables: x12, x14
    2. Equation: y3 * -1.0
6. Create design variables links for the 60 degree plies
    1. Dependent Design Variables: x11, x13
    2. Equation: y3 * 1.0

- It is important to verify the Equation is configured properly. For example, the variable x10 is initially equal to -85 degrees. When the Equation is configured, it should also produce an initial value of -85. The resulting value of the Equation is displayed on the column titled Value of Equation and can be used to validate the Equation is configured properly.

Step 3 - Create variable links

1 + Create DLINK

+ Options

	Status	Dependent Design Variables	Equation (Independent Design Variables)	Value of Equation
	Search	Search	Search	Search
✖	✔	x1, x2, x3, x4, x5, x6, x7, x8 2.1	y1 * 1.0 2.2	0.01
✖	✔	x10, x16 3.1	y2 * -1.0 3.2	-85.
✖	✔	x9, x15 4.1	y2 * 1.0 4.2	85.
✖	✔	x12, x14 5.1	y3 * -1.0 5.2	-60.
✖	✔	x11, x13 6.1	y3 * 1.0 6.2	60.

DLINK Entries

DLINK entries are used to create linear relationships between variables.

The SOL 200 Web App allows multiple variations of inputting the linear relationships.

Ultimately, the relationships result in one specific format. To the right are the equivalent linear relationships for the image shown above.

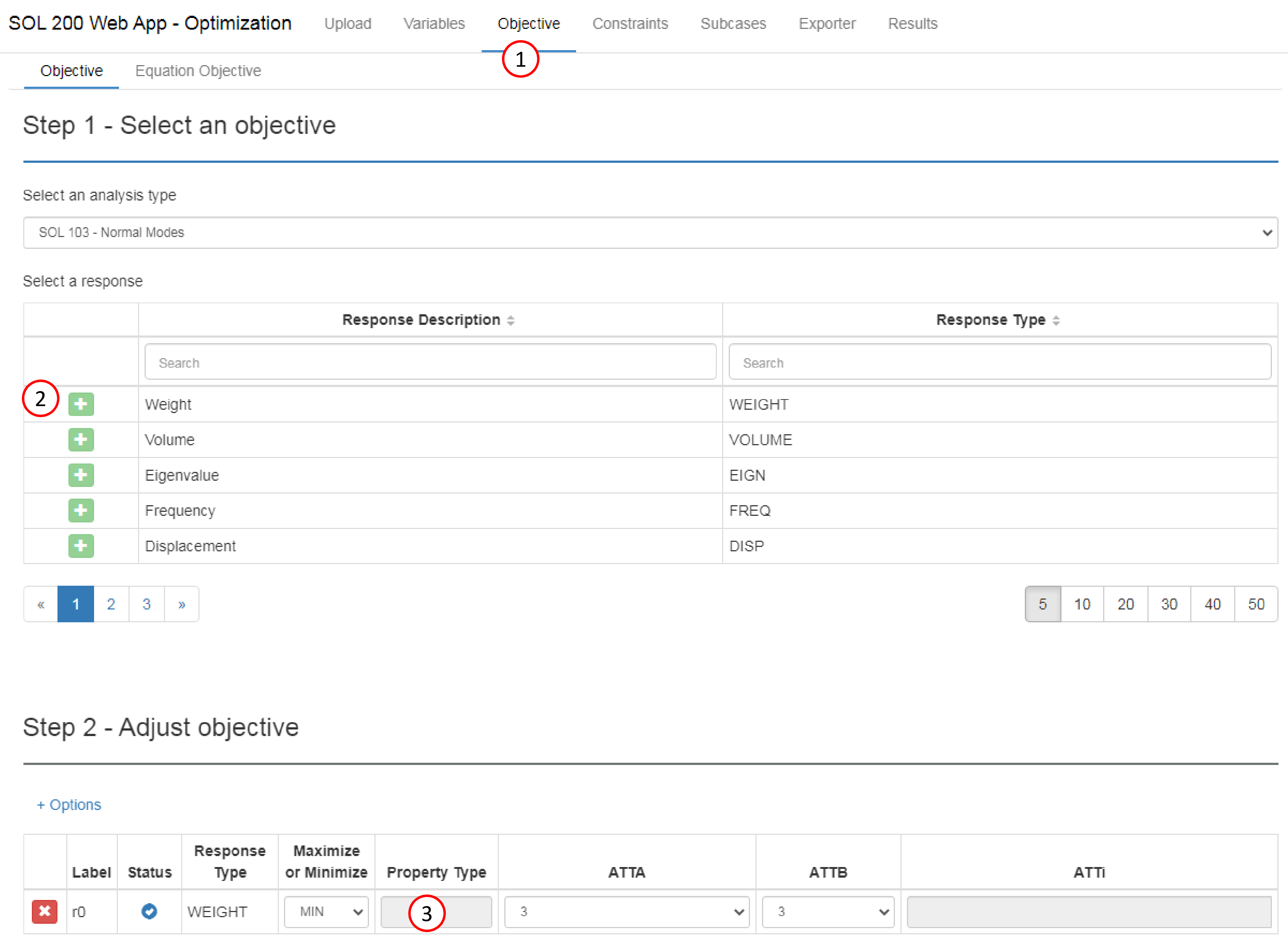
The right of the expression can also have additional variables. For example, x2 = 1.5 + x1 * 1.0 + y2 * -3.5 + ...

x1 = 0.0 + y1 * 1.0
x2 = 0.0 + y1 * 1.0
...
x8 = 0.0 + y1 * 1.0
x9 = 0.0 + y2 * 1.0
x10 = 0.0 + y2 * -1.0
x16 = 0.0 + y2 * -1.0
x15 = 0.0 + y2 * 1.0
x11 = 0.0 + y3 * 1.0
x12 = 0.0 + y3 * -1.0
x13 = 0.0 + y3 * 1.0
x14 = 0.0 + y3 * -1.0

Create Design Objective

1. Click Objective
2. Select the plus (+) icon for weight
3. The objective has been set to minimize the weight, no further modification is necessary

- The objective must always be a single and global response. A response such as weight and volume are single responses, are independent of load case, and can be used as an objective. Other responses require special care when set as an objective. For example, if the objective is stress, only the stress of a single component, e.g. von Mises, of a single element, of a single load case may be used.

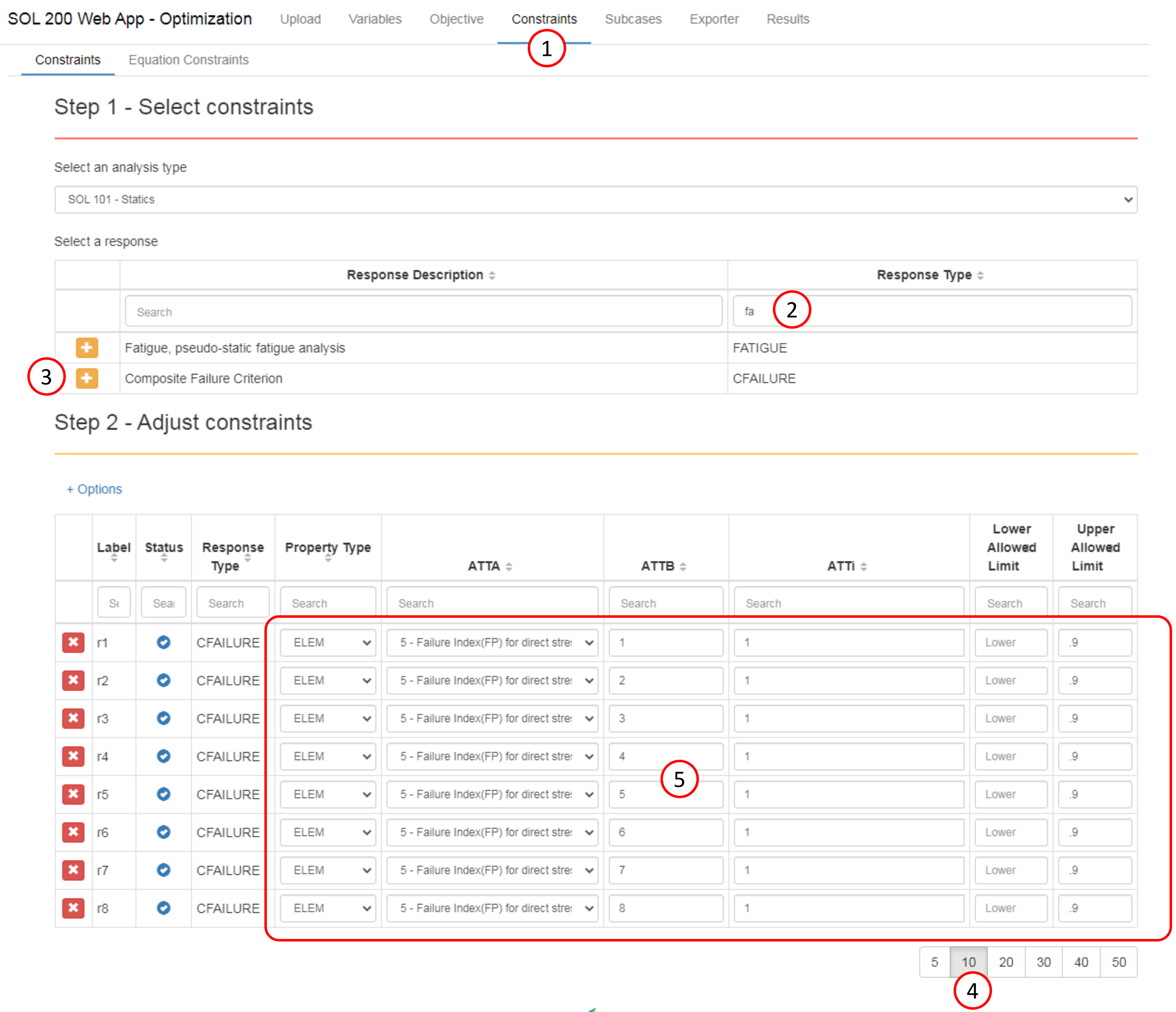


Create Design Constraints

1. Click Constraints
2. In the search box, type 'fa'
3. Select the plus(+) icon 8 times for Composite Failure Criterion to create 8 constraints
4. Click 10 on the pagination bar
5. Configure the constraints as shown to the right

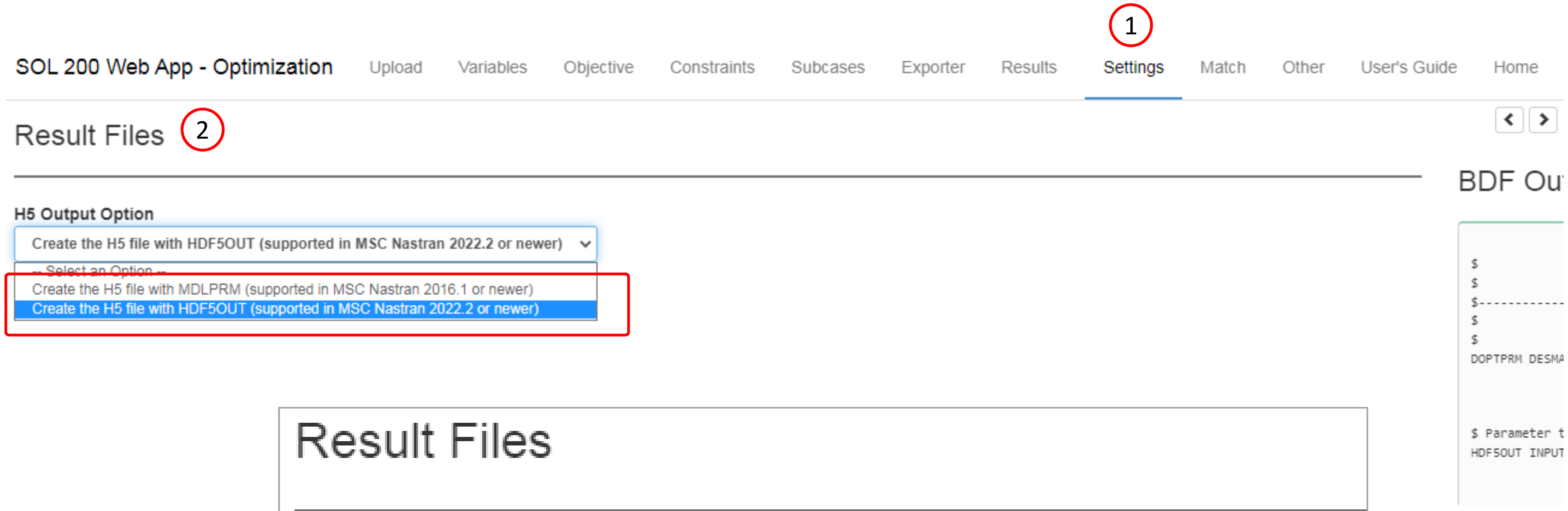
- Example: Configure the following for r1
  - Property Type: ELEM
  - ATTA: 5 - FP (failure index)
  - ATTB: 1 (lamina 1)
  - ATTi: 1 (element 1)
  - Upper Allowed Limit: .9
- Repeat the same for r2, r3, ... r8, but note that ATTB will be different for each row

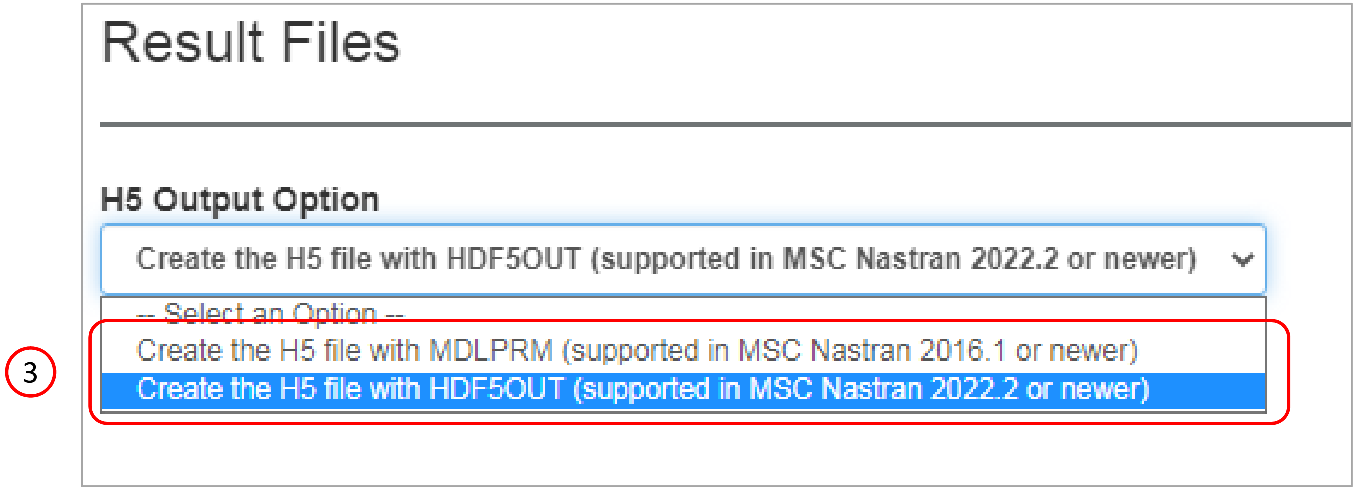
- This example requires 8 constraints to be created. If there is a need to create hundreds of constraints, the web app includes a CSV export/import capability and is available by clicking + Options. With the use of CSV and Excel, constraints can be quickly generated.
- In prior versions of this tutorial, the Lower Allowed Limit was set to .001. The failure index can actually be lower, e.g. .0002, and is still valid, but with the lower bound set to .001, the optimizer would see .0002 as a violated constraint. In this version of the tutorial, the lower allowed limit is left blank to avoid this issue.



Configure Settings

1. Click Settings
2. Scroll to section Result Files
3. Select one of the following H5 output options
    - Create the H5 file with MDLPRM
    - Create the H5 file with HDF5OUT



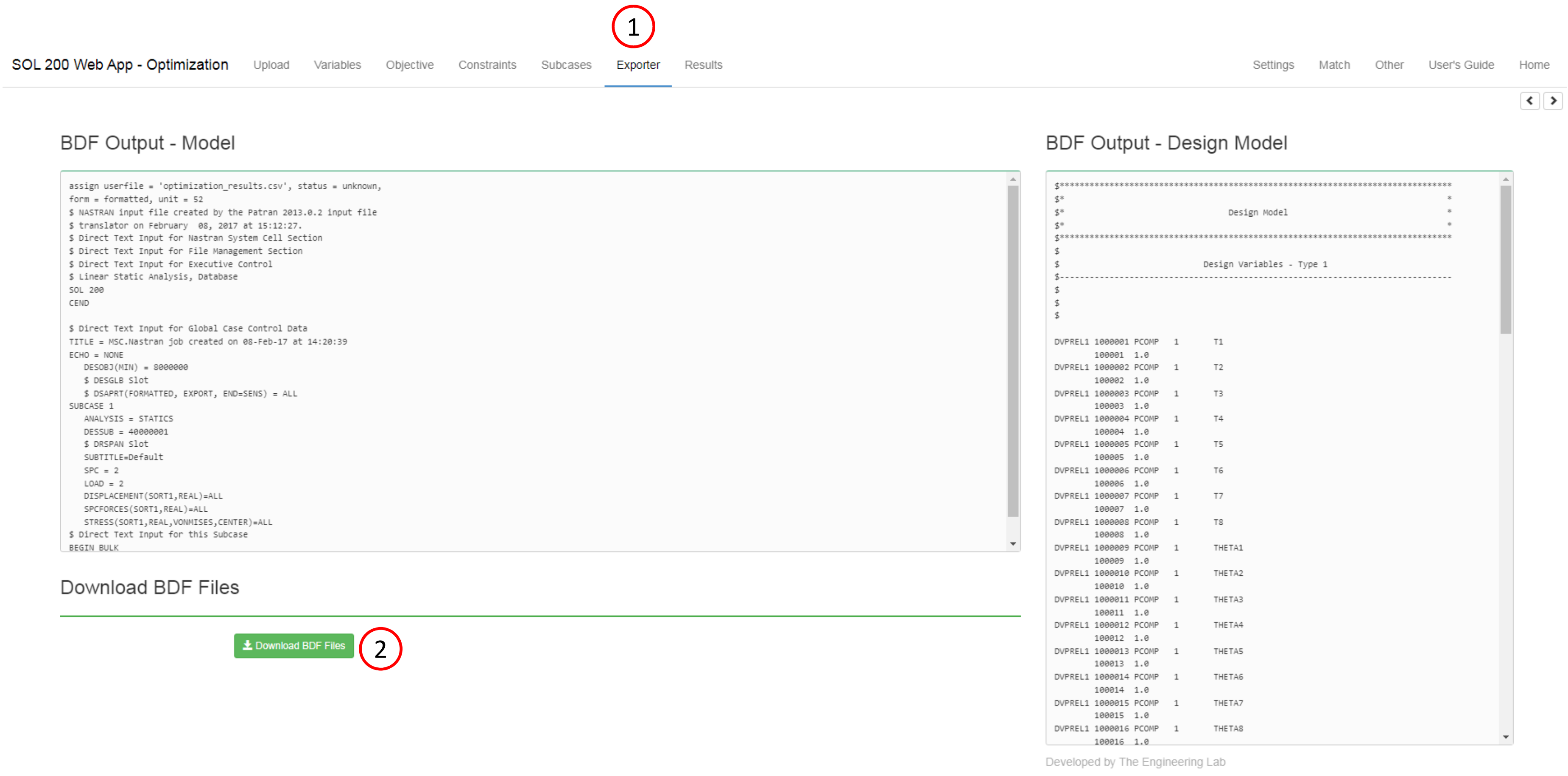


- The H5 file is used by the Post-processor web app to display MSC Nastran results.
- The H5 file is used by the HDF5 Explorer to create graphs (XY Plots) of MSC Nastran results.

Export New BDF Files

1. Click on Exporter
2. Click on Download BDF Files

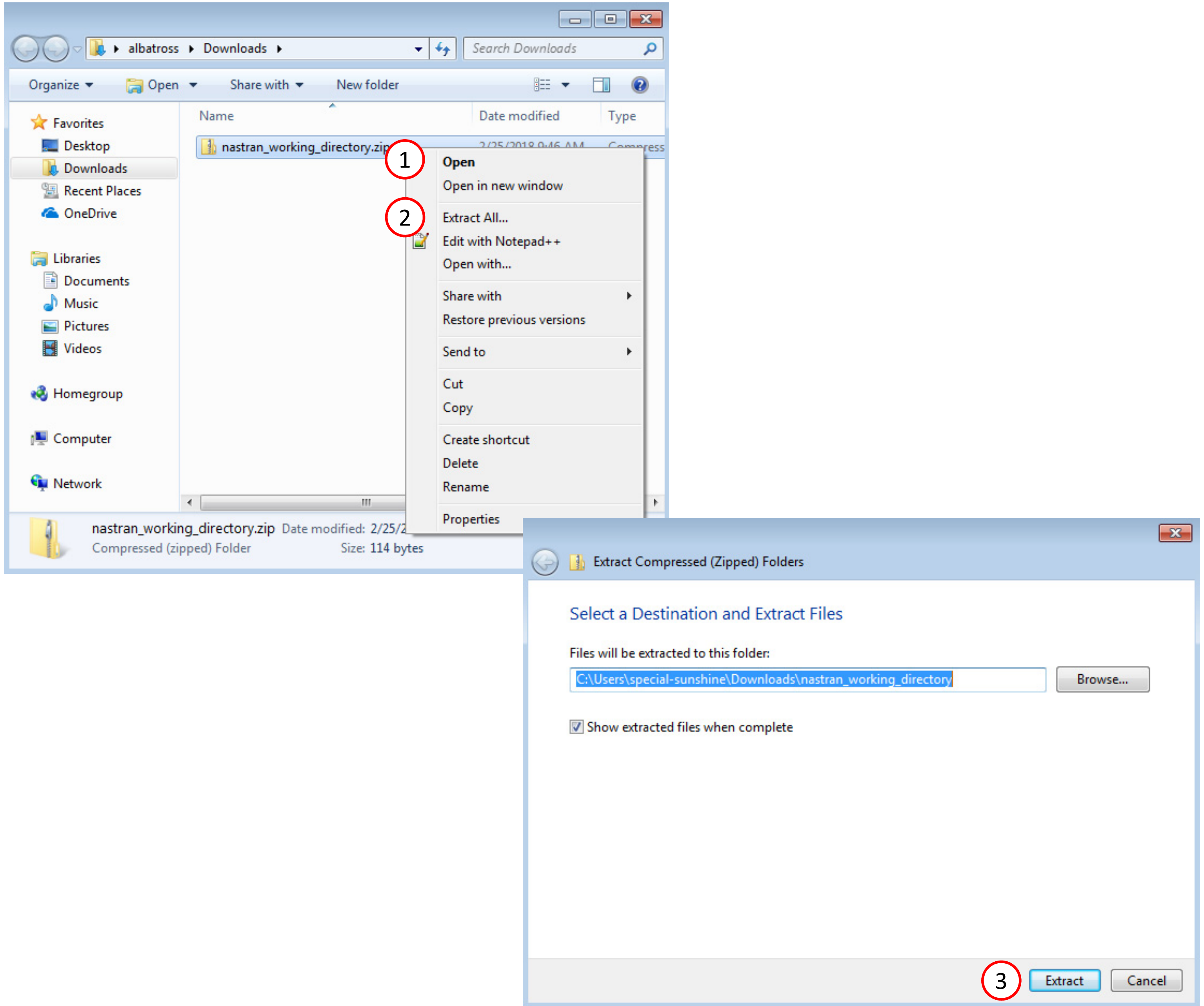
- When the download button is clicked a new file named “nastran_working_directory” is downloaded. If the file already exists in your local folder, the folder name is appended with a number, e.g. “nastran_working_directory (1).zip”



Perform the Optimization with Nastran SOL 200

1. A new .zip file has been downloaded
2. Right click on the file
3. Click Extract All
4. Click Extract on the following window

- Always extract the contents of the ZIP file to a new, empty folder.



Perform the Optimization with Nastran SOL 200

1. Inside of the new folder, double click on Start MSC Nastran
2. Click Open, Run or Allow Access on any subsequent windows
3. MSC Nastran will now start

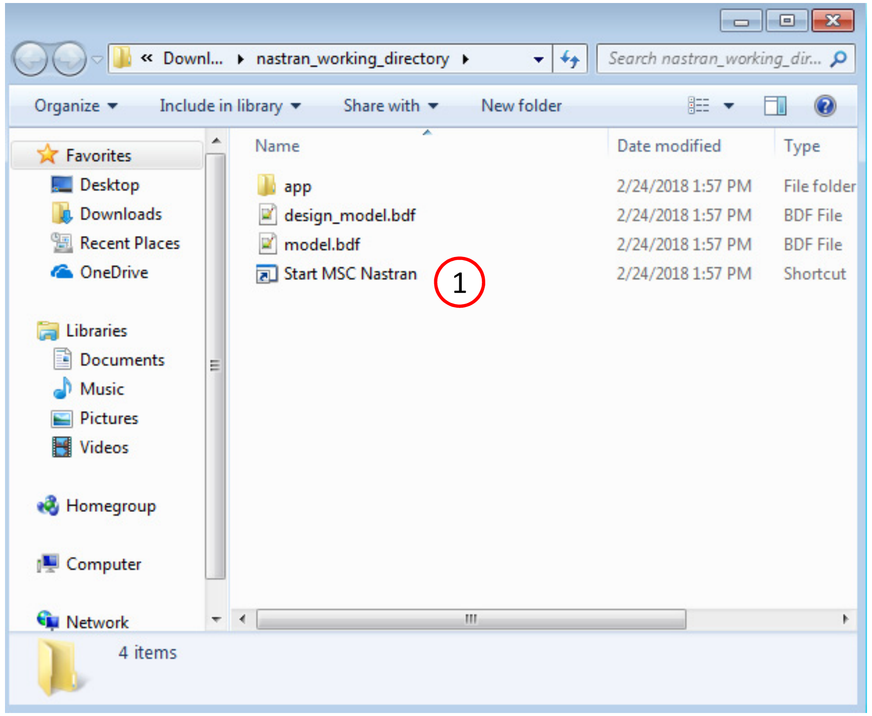
- After a successful optimization, the results will be automatically displayed as long as the following files are present: BDF, F06 and LOG.
- One can run the Nastran job on a remote machine as follows:
  1) Copy the BDF files and the INCLUDE files to a remote machine. 2) Run the MSC Nastran job on the remote machine. 3) After completion, copy the BDF, F06, LOG, H5 files to the local machine. 4) Click "Start MSC Nastran" to display the results.

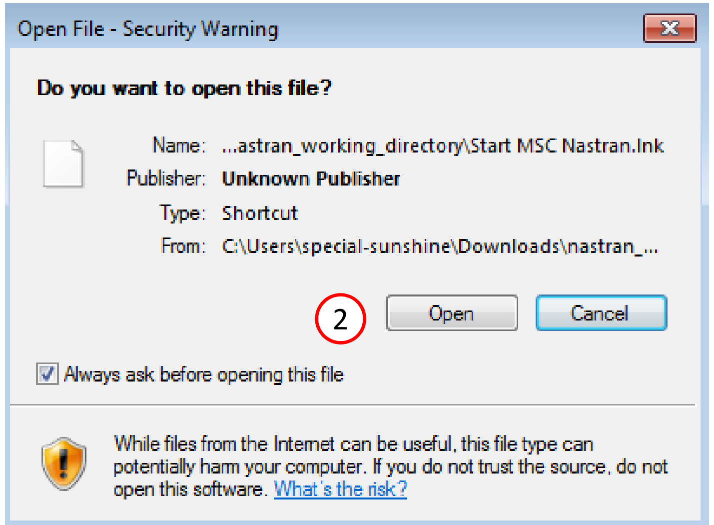
Using Linux?

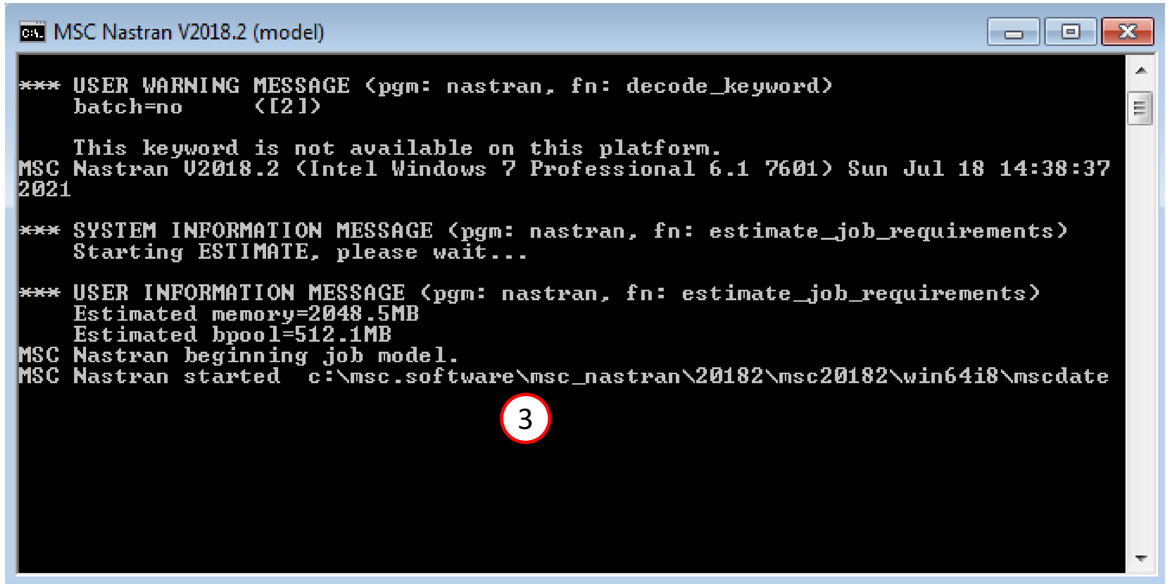
Follow these instructions:
1) Open Terminal
2) Navigate to the nastran_working_directory
cd ./nastran_working_directory
3) Use this command to start the process
./Start_MSC_Nastran.sh

In some instances, execute permission must be granted to the directory. Use this command. This command assumes you are one folder level up.

sudo chmod -R u+x ./nastran_working_directory







Status

1. While MSC Nastran is running, a status page will show the current state of MSC Nastran

- The status of the MSC Nastran job is reported on the Status page. Note that Windows 7 users will experience a delay in the status updates. All other users of Windows 10 and Red Hat Linux will see immediate status updates.

SOL 200 Web App - Status

Python MSC Nastran

Status

Name	Status of Job	Design Cycle	RUN TERMINATED DUE TO
model.bdf	Running	None	

Review Optimization Results

After MSC Nastran is finished, the results will be automatically uploaded.

1. Ensure the messages shown have green checkmarks. This is indication of success. Any red icons indicate challenges.

2. The final value of objective, normalized constraints (not shown) and design variables can be reviewed.

- Both the thicknesses and orientation angles have changed during the optimization. The plot can be manipulated to view specific variables.
- On some occasions, the orientation angles may not change during the optimization. See the Appendix for additional details.

Results

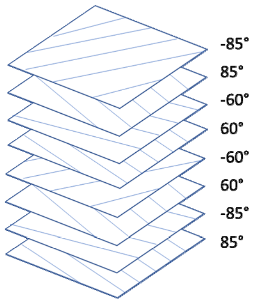
Objective: Minimize the weight of this cylinder composed of a composite laminate

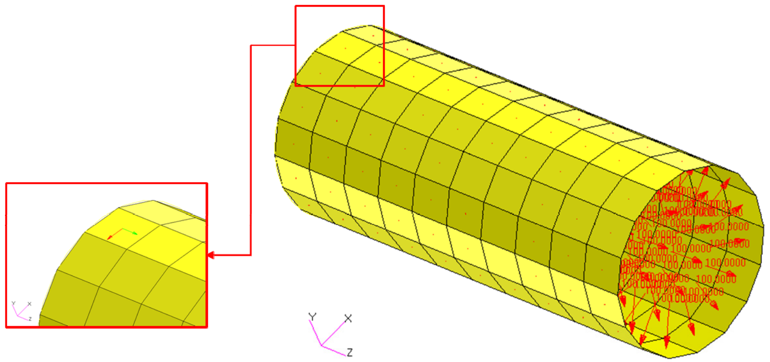
Before Optimization

- Weight: 1.60 $lb_f \cdot s^2/in$
- Layup: [85/-85/60/-60/60/-60/85/-85]
- Thickness: .0100 in
- Plies are initially in failure

After Optimization

- Weight: 1.04 $lb_f \cdot s^2/in$
- Layup: [90/-90/0/0/0/0/90/-90]
- Thickness: .0065 in





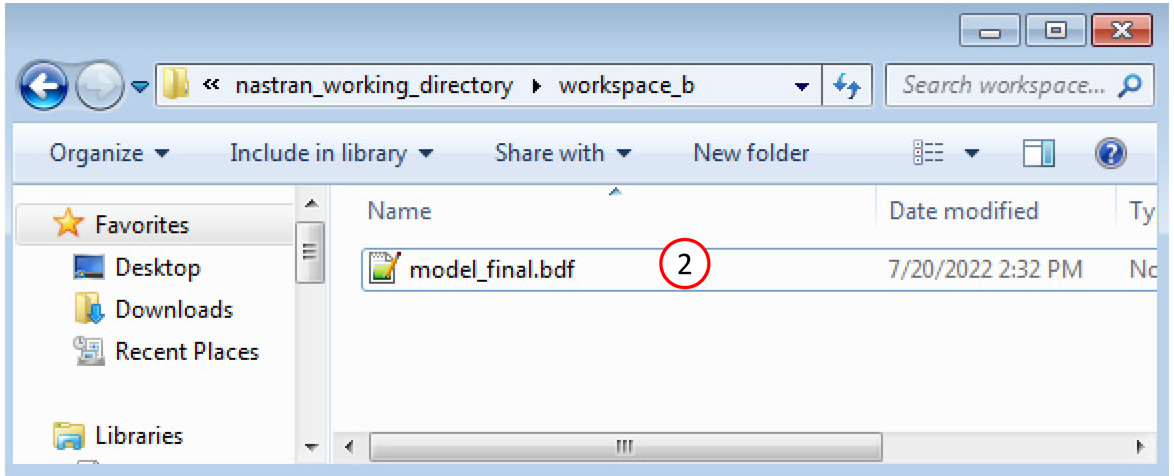
Update the Original Model

1. The original input files, e.g. DAT, BDF, etc., contains the original values for the designed properties. These original values must be updated to use the new and optimized values.

2. A new BDF file has been created in nastran_working_directory/workspace_b/model_final.bdf.

3. The file model_final.bdf is a copy of the original input files but the original values for the designed properties have been updated to use the optimized values.

- If you were using multiple INCLUDE files, model_final.bdf is a combination of all INCLUDE files. The next few slides discuss an alternative method of using the PCH to BDF web app to update the values for the designed properties while preserving separate INCLUDE files.



Original Input Files

1

```
$ Composite Material Description :
PCOMP    1                           13000.   HILL
              1     .01     85.     YES     1      .01     -85.       YES
              1     .01     60.     YES     1      .01     -60.       YES
              1     .01     60.     YES     1      .01     -60.       YES
              1     .01     85.     YES     1      .01     -85.       YES
$ Pset: "composite_tube" will be imported as: "pcomp.1"
CQUAD4   1      1       1       2       19      18
```

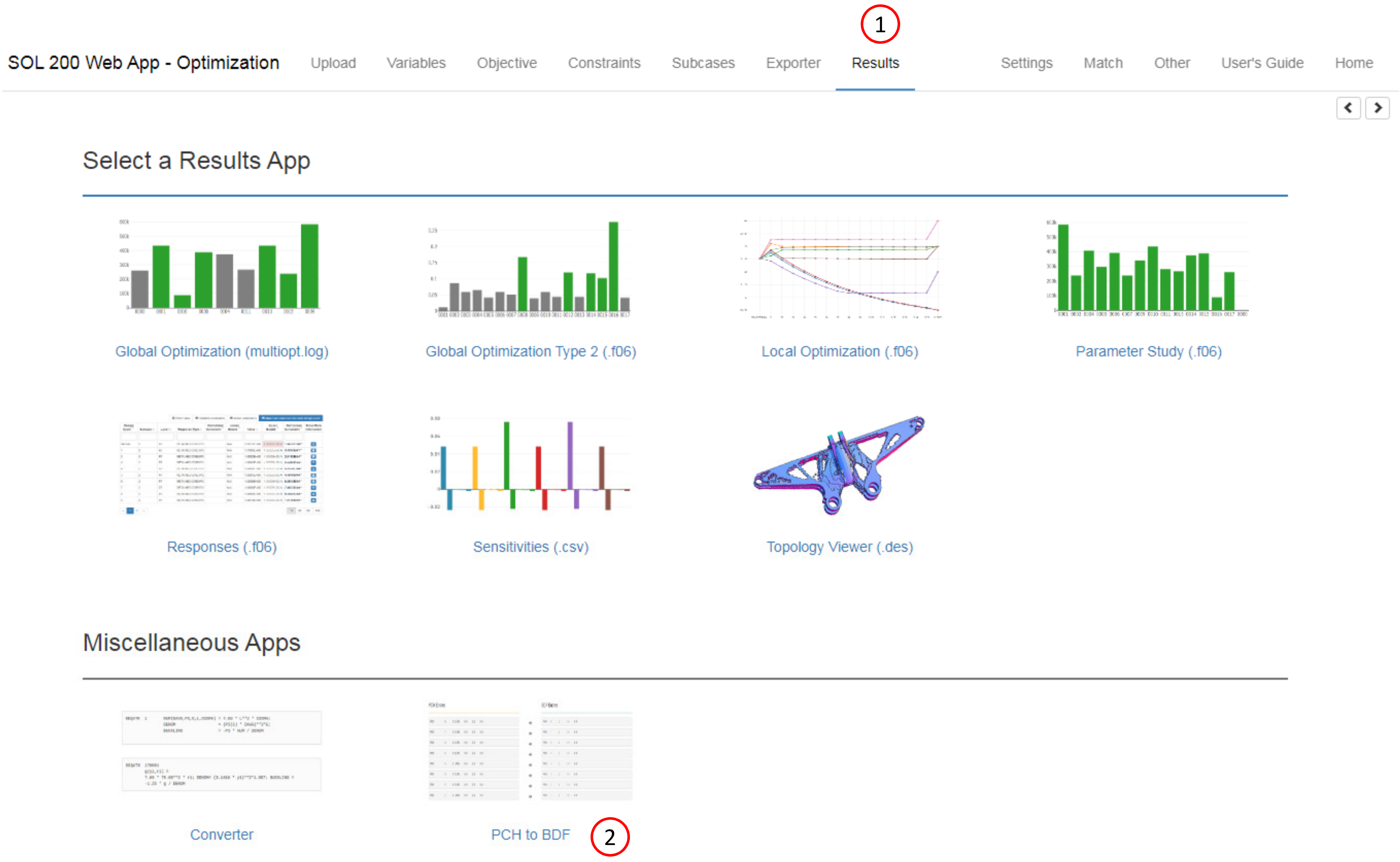
Updated BDF File (model_final.bdf)

3

```
$ Composite Material Description :
PCOMP           1          0.0  13000.     HILL   0.0       0.0
                1 .00652  90.          YES          1 .00652 -90.         YES
                1 .00652    0.0        YES          1 .00652    0.0       YES
                1 .00652    0.0        YES          1 .00652    0.0       YES
                1 .00652  90.          YES          1 .00652 -90.         YES
$ Pset: "composite_tube" will be imported as: "pcomp.1"
CQUAD4   1        1        1        2        19       18
```

Update the Original Model

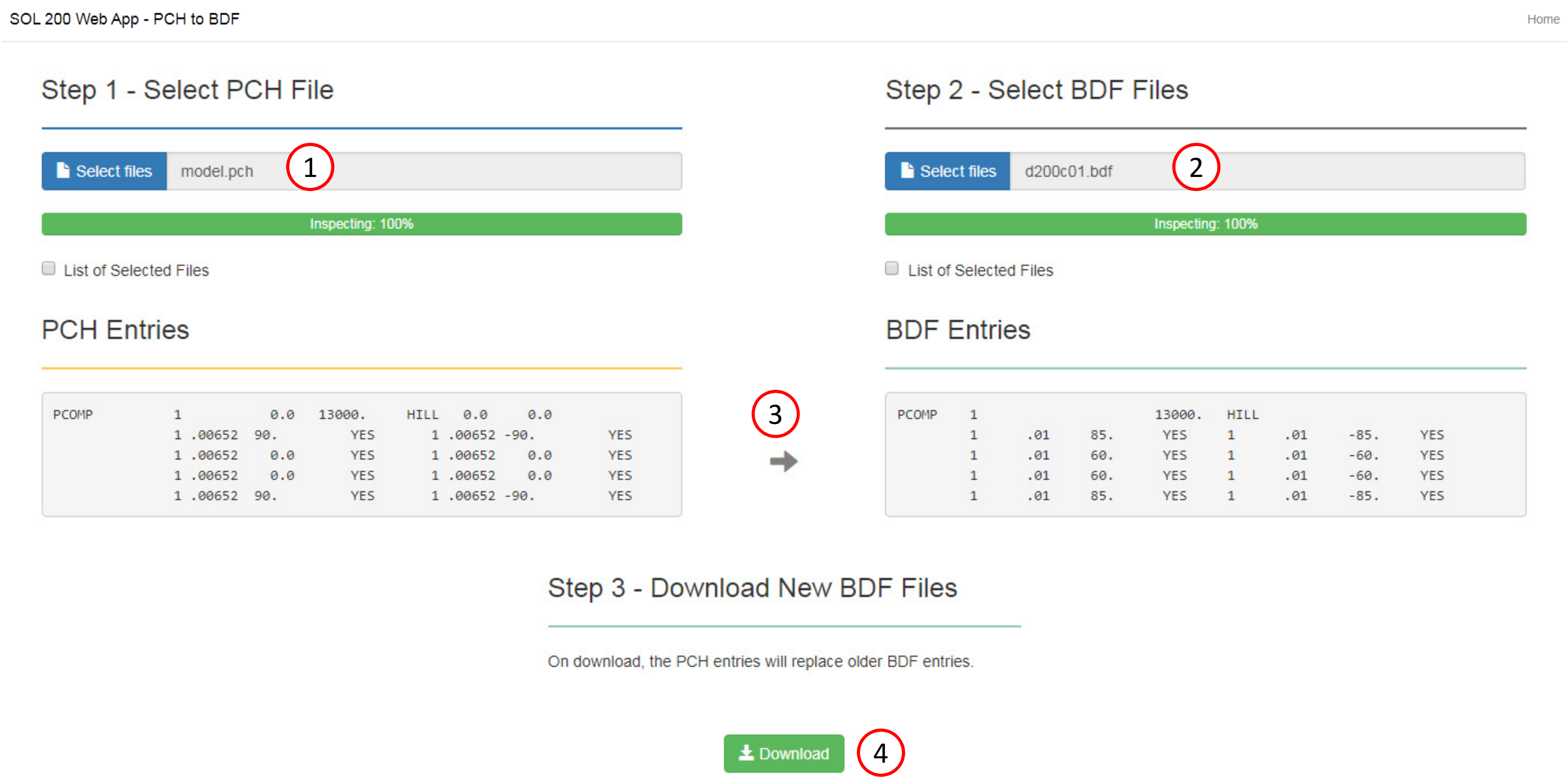
1. Click Results
2. Click PCH to BDF



Update the Original Model

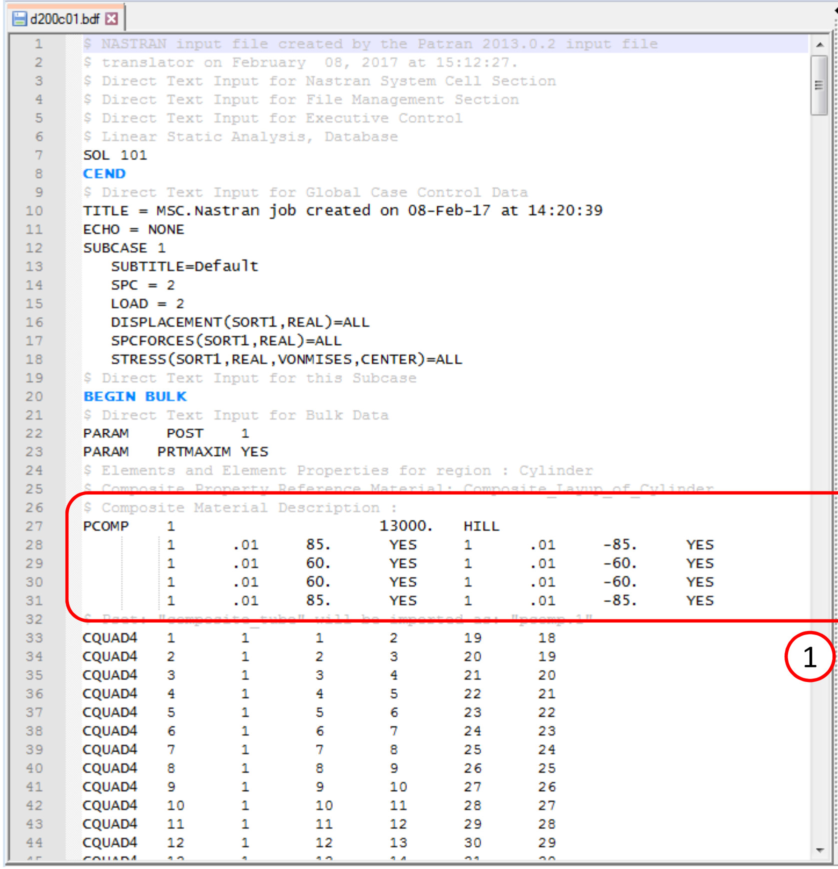
The original .bdf/.dat file has old information about the properties. The properties will be updated.

1. Select the model.pch file
2. Select the original file: d200c01.bdf
3. A summary of updates that will be performed are shown
4. Click Download and a new updated BDF file is downloaded



Update the Original Model

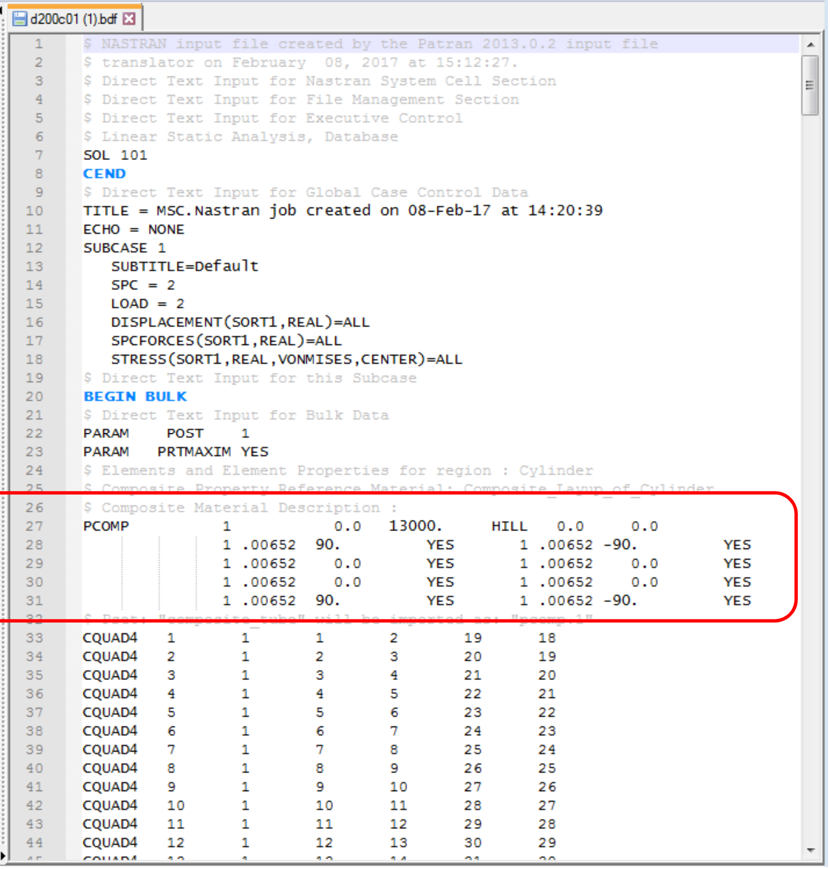
1. Note the entries have been updated with the optimized properties



d200c01.bdf

```
$ NASTRAN input file created by the Patran 2013.0.2 input file
$ translator on February  08, 2017 at 15:12:27.
$ Direct Text Input for Nastran System Cell Section
$ Direct Text Input for File Management Section
$ Direct Text Input for Executive Control
$ Linear Static Analysis, Database
SOL 101
CEND
$ Direct Text Input for Global Case Control Data
TITLE = MSC.Nastran job created on 08-Feb-17 at 14:20:39
ECHO = NONE
SUBCASE 1
   SUBTITLE=Default
   SPC = 2
   LOAD = 2
   DISPLACEMENT(SORT1,REAL)=ALL
   SPCFORCES(SORT1,REAL)=ALL
   STRESS(SORT1,REAL,VONMISES,CENTER)=ALL
$ Direct Text Input for this Subcase
BEGIN BULK
$ Direct Text Input for Bulk Data
PARAM    POST    1
PARAM   PRTMAXIM YES
$ Elements and Element Properties for region : Cylinder
$ Composite Property Reference Material: Composite_Layup_of_Cylinder
$ Composite Material Description :
PCOMP    1                      13000.  HILL
         1       .01     85.     YES     1       .01     -85.    YES
         1       .01     60.     YES     1       .01     -60.    YES
         1       .01     60.     YES     1       .01     -60.    YES
         1       .01     85.     YES     1       .01     -85.    YES
$ Pset: "composite_tube" will be imported as: "pcomp.1"
CQUAD4   1       1       1       2       19      18
CQUAD4   2       1       2       3       20      19
CQUAD4   3       1       3       4       21      20
CQUAD4   4       1       4       5       22      21
CQUAD4   5       1       5       6       23      22
CQUAD4   6       1       6       7       24      23
CQUAD4   7       1       7       8       25      24
CQUAD4   8       1       8       9       26      25
CQUAD4   9       1       9       10      27      26
CQUAD4   10      1       10      11      28      27
CQUAD4   11      1       11      12      29      28
CQUAD4   12      1       12      13      30      29
```

Original BDF/DAT File



d200c01 (1).bdf

```
$ NASTRAN input file created by the Patran 2013.0.2 input file
$ translator on February  08, 2017 at 15:12:27.
$ Direct Text Input for Nastran System Cell Section
$ Direct Text Input for File Management Section
$ Direct Text Input for Executive Control
$ Linear Static Analysis, Database
SOL 101
CEND
$ Direct Text Input for Global Case Control Data
TITLE = MSC.Nastran job created on 08-Feb-17 at 14:20:39
ECHO = NONE
SUBCASE 1
   SUBTITLE=Default
   SPC = 2
   LOAD = 2
   DISPLACEMENT(SORT1,REAL)=ALL
   SPCFORCES(SORT1,REAL)=ALL
   STRESS(SORT1,REAL,VONMISES,CENTER)=ALL
$ Direct Text Input for this Subcase
BEGIN BULK
$ Direct Text Input for Bulk Data
PARAM    POST    1
PARAM   PRTMAXIM YES
$ Elements and Element Properties for region : Cylinder
$ Composite Property Reference Material: Composite_Layup_of_Cylinder
$ Composite Material Description :
PCOMP          1     0.0 13000.     HILL    0.0     0.0
               1 .00652 90.         YES       1 .00652 -90.         YES
               1 .00652    0.0      YES       1 .00652    0.0       YES
               1 .00652    0.0      YES       1 .00652    0.0       YES
               1 .00652 90.         YES       1 .00652 -90.         YES
$ Pset: "composite_tube" will be imported as: "pcomp.1"
CQUAD4   1       1       1       2       19      18
CQUAD4   2       1       2       3       20      19
CQUAD4   3       1       3       4       21      20
CQUAD4   4       1       4       5       22      21
CQUAD4   5       1       5       6       23      22
CQUAD4   6       1       6       7       24      23
CQUAD4   7       1       7       8       25      24
CQUAD4   8       1       8       9       26      25
CQUAD4   9       1       9       10      27      26
CQUAD4   10      1       10      11      28      27
CQUAD4   11      1       11      12      29      28
CQUAD4   12      1       12      13      30      29
```

Downloaded BDF/DAT File

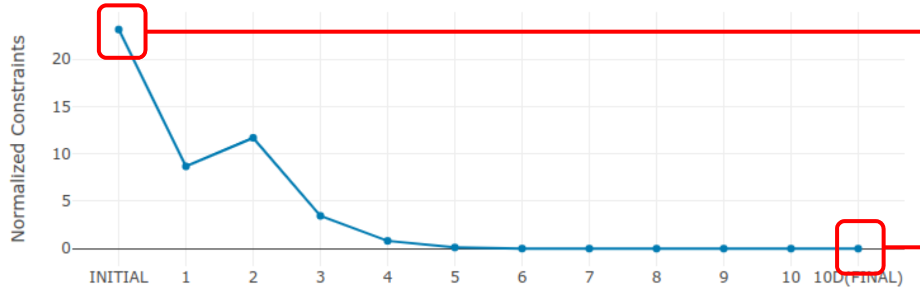
Inspection of MSC Nastran Results with the Post-processor Web App

Normalized Constraints

- All constraints are normalized. For each design cycle, the maximum normalized constraint (NC) is reported in the Normalized Constraints plot.
- The Responses web app is used to inspect the corresponding response for each maximum normalized constraint value.
    - For the initial design, the maximum NC is 23.135 and corresponds to a failure index of 21.721.
    - For the final design, the maximum NC is -.00018746 and corresponds to a failure index of 89.983.

Normalized Constraints

+ Info



SOL 200 Web App - Responses

Home

Responses

Reset view | Violated constraints | Active constraints | Maximum constraint for each design cycle

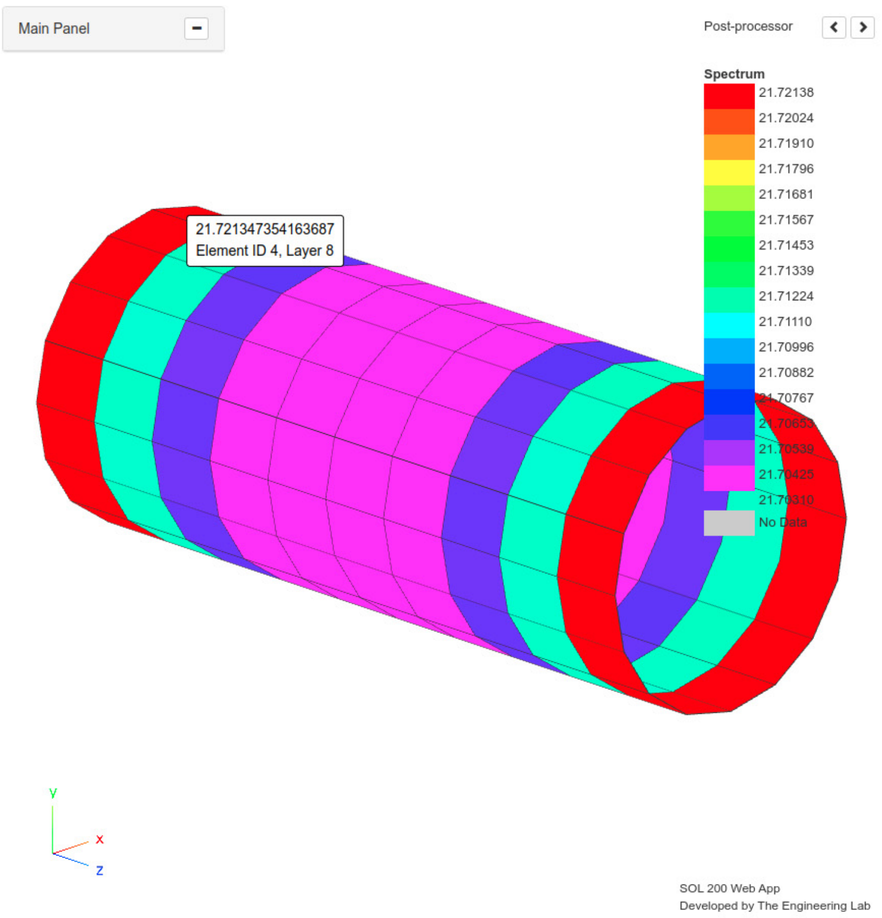
Design Cycle	Subcase	Label	Response Type	Normalized Constraint	Lower Bound	Value	Upper Bound	Normalized Constraint	Show More Information	designCycleNumber	ELEMENT ID	COMPONENT NO.	LAMINA NO.
Search	Searc	S	Search	Search	Sea	Search	Search	Search					
INITIAL	1	r8	LAMINA FAILURE INDEX		N/A	2.1721E+01	9.0000E-01 V	**2.3135E+01****		0	1	5	8
1	1	r8	LAMINA FAILURE INDEX		N/A	8.7169E+00	9.0000E-01 V	**8.6855E+00****		1	1	5	8
2	1	r8	LAMINA FAILURE INDEX		N/A	1.1430E+01	9.0000E-01 V	**1.1700E+01****		2	1	5	8
3	1	r8	LAMINA FAILURE INDEX		N/A	4.0147E+00	9.0000E-01 V	**3.4608E+00****		3	1	5	8
4	1	r8	LAMINA FAILURE INDEX		N/A	1.6389E+00	9.0000E-01 V	**8.2099E-01****		4	1	5	8
5	1	r8	LAMINA FAILURE INDEX		N/A	1.0114E+00	9.0000E-01 V	**1.2382E-01****		5	1	5	8
6	1	r8	LAMINA FAILURE INDEX		N/A	9.0861E-01	9.0000E-01 V	**9.5659E-03****		6	1	5	8
7	1	r8	LAMINA FAILURE INDEX		N/A	9.0465E-01	9.0000E-01 V	**5.1700E-03****		7	1	5	8
8	1	r8	LAMINA FAILURE INDEX		N/A	9.0121E-01	9.0000E-01 A	**1.3412E-03****		8	1	5	8
9	1	r8	LAMINA FAILURE INDEX		N/A	9.0161E-01	9.0000E-01 A	**1.7926E-03****		9	1	5	8
FINAL - 10D(FI	1	r8	LAMINA FAILURE INDEX		N/A	8.9983E-01	9.0000E-01 A	**-1.8746E-04****		10	1	5	8

Post-processor Web App

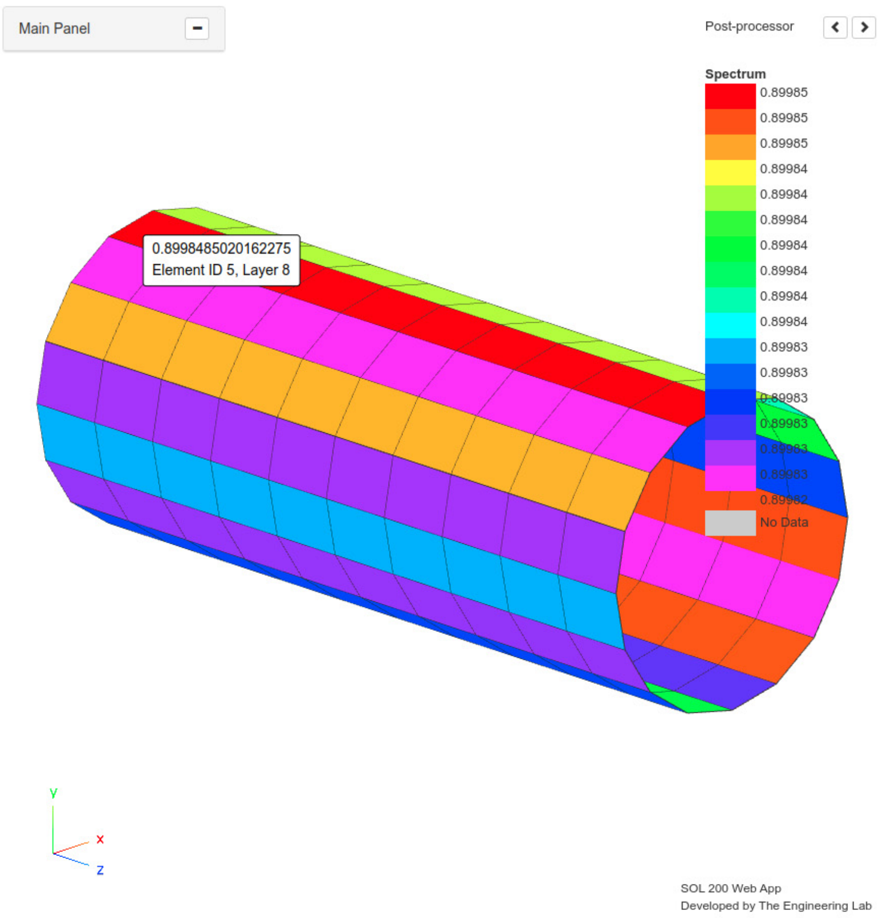
- The Post-processor web app is used to inspect the maximum failure index for all layers.
- For the initial design, the maximum failure index is 21.7213.
- For the final design, the maximum failure index is .8998.

- Refer to the Post-processor web app tutorials to learn more about MSC Nastran results.

Fringe Plot: Max Failure Index



Initial Design



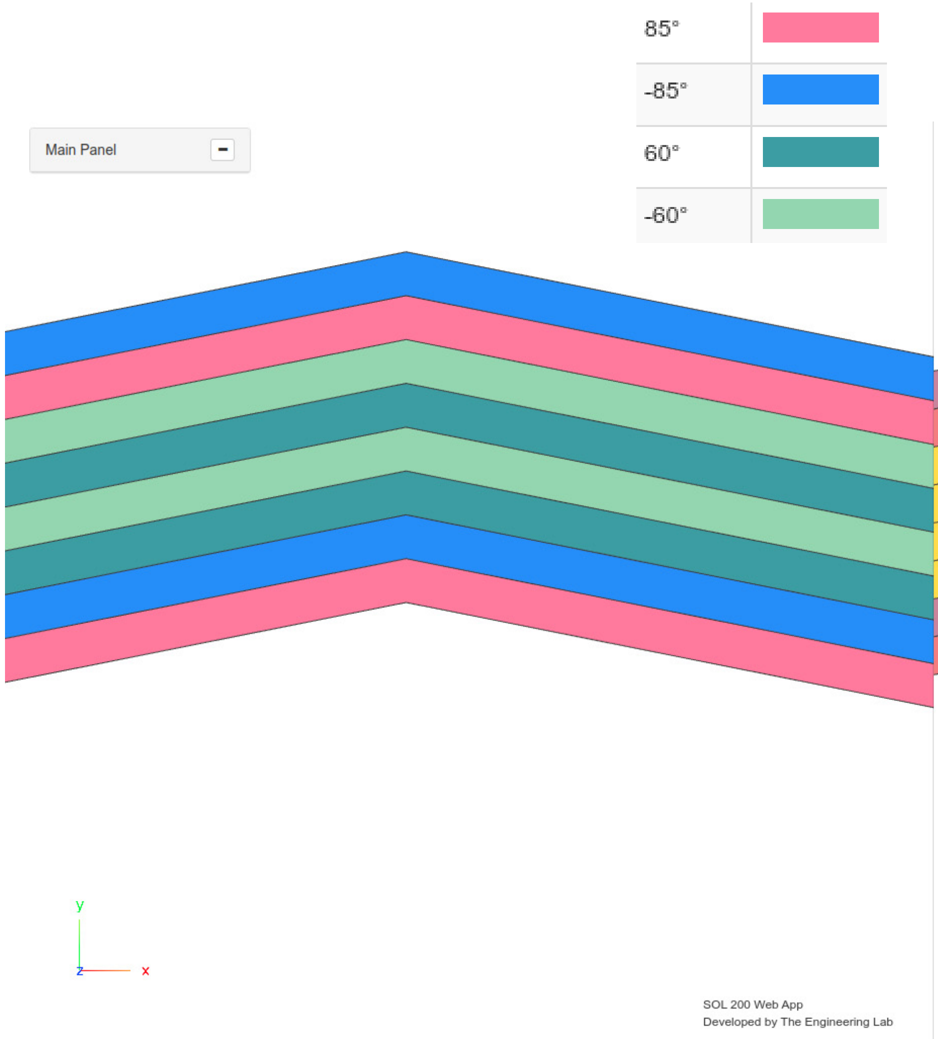
Final Design

Post-processor Web App

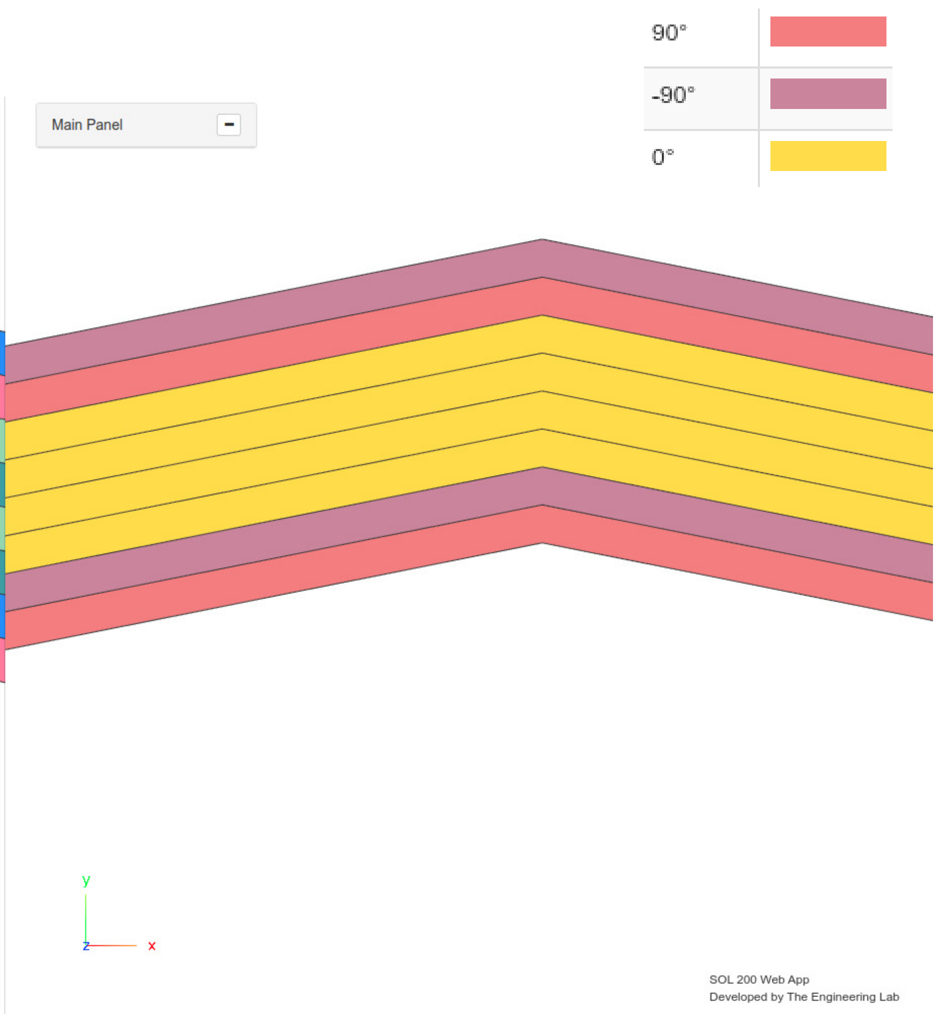
- The layer thicknesses of the initial and final design are compared.

- Refer to the Post-processor web app tutorials to learn more about MSC Nastran results.

Thickness of PCOMP Layers



Initial Design



Final Design

End of Tutorial

Appendix

Appendix Contents

- Frequently Asked Questions
  - What are the ways of specifying allowable discrete values for design variables?
  - Why do the orientation angles not change?

Frequently Asked Questions

Question:

- What are the ways of specifying allowable discrete values for design variables?

Answer:

- There are two methods
- Suppose you want to specify allowable values: 1.0, 2.0, 3.0, …. 10.0

Method 1

- Type in each value with commas separating each value

Allowed Values

1.0, 2.0, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0, 9.0, 10.0

```
DDVAL   2001
        1.0     2.0     3.0     4.0     5.0     6.0     7.0     8.0
        9.0     10.0
```

Method 2

- Use the THRU and BY method

Allowed Values

1.0, 2.0, 3.0, 4.0, 5.0, THRU, 10., BY, 1.0

Specifies range of values

Specifies increment between values

```
DDVAL   2001
        1.0     2.0     3.0     4.0     5.0     THRU    10.     BY
        1.0
```

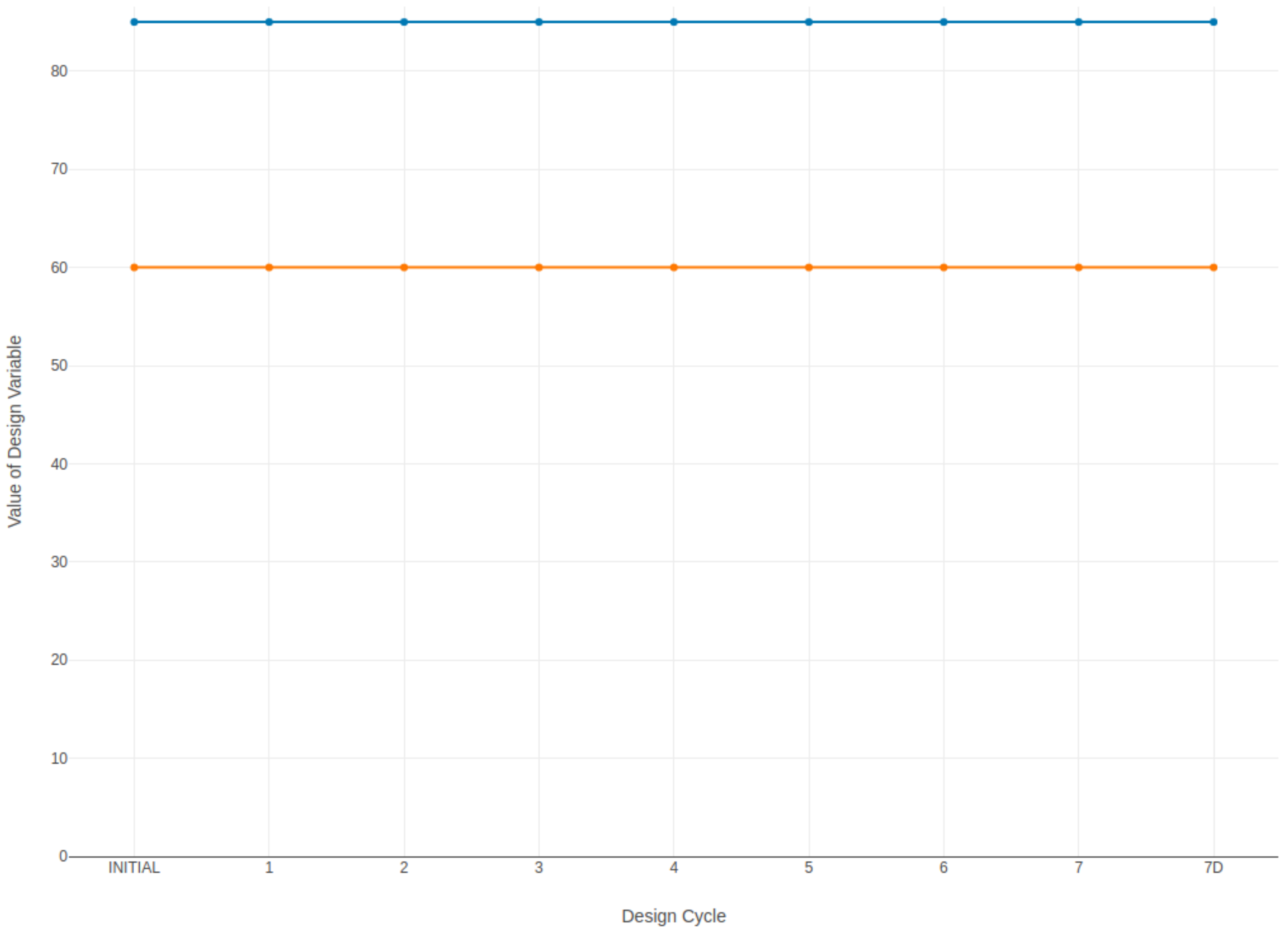
Why do the orientation angles not change?

Why do the orientation angles not change?

For some optimizations, the orientation angles remain unchanged.

Design Variables

	x9	✔	THETA1, Orientation angle of ply, of PCOMP 1 (Cylinder)
	x11	✔	THETA3, Orientation angle of ply, of PCOMP 1 (Cylinder)



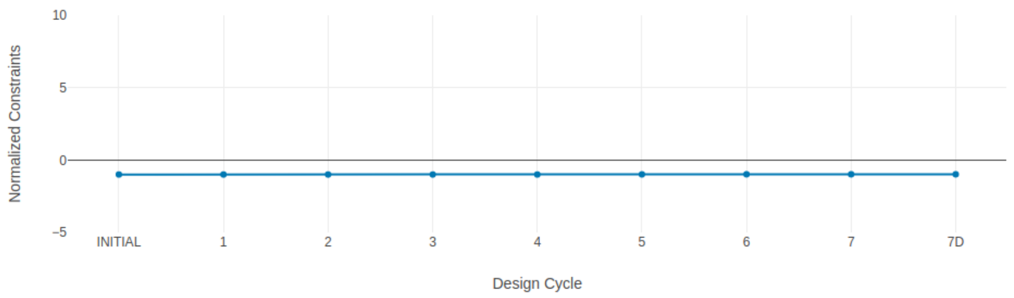
Why do the orientation angles not change?

The orientation angles will only change when an angle dependent constraint **<u>is violated</u>**.

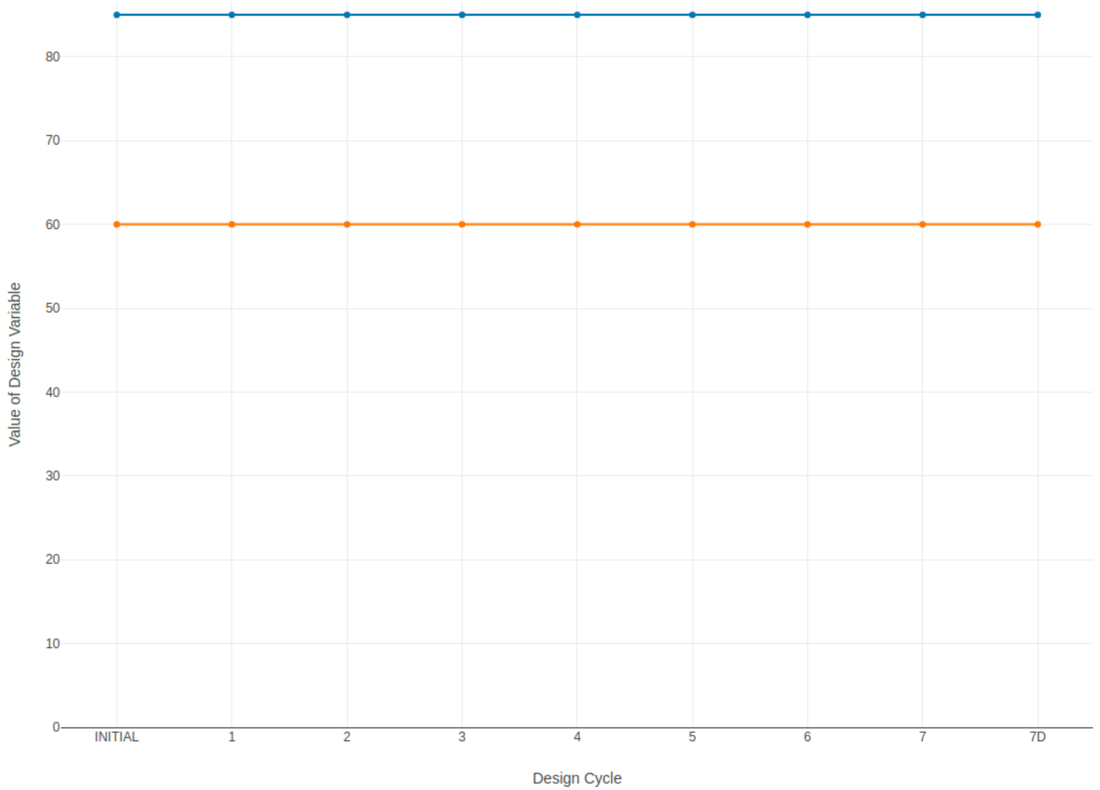
In this example, the normalized constraint at each design cycle is negative, indicating satisfied constraints. There are no violated constraints, i.e. positive normalized constraints, therefore, there is no change in orientation angles.

Normalized Constraints

+ Info



Design Variables

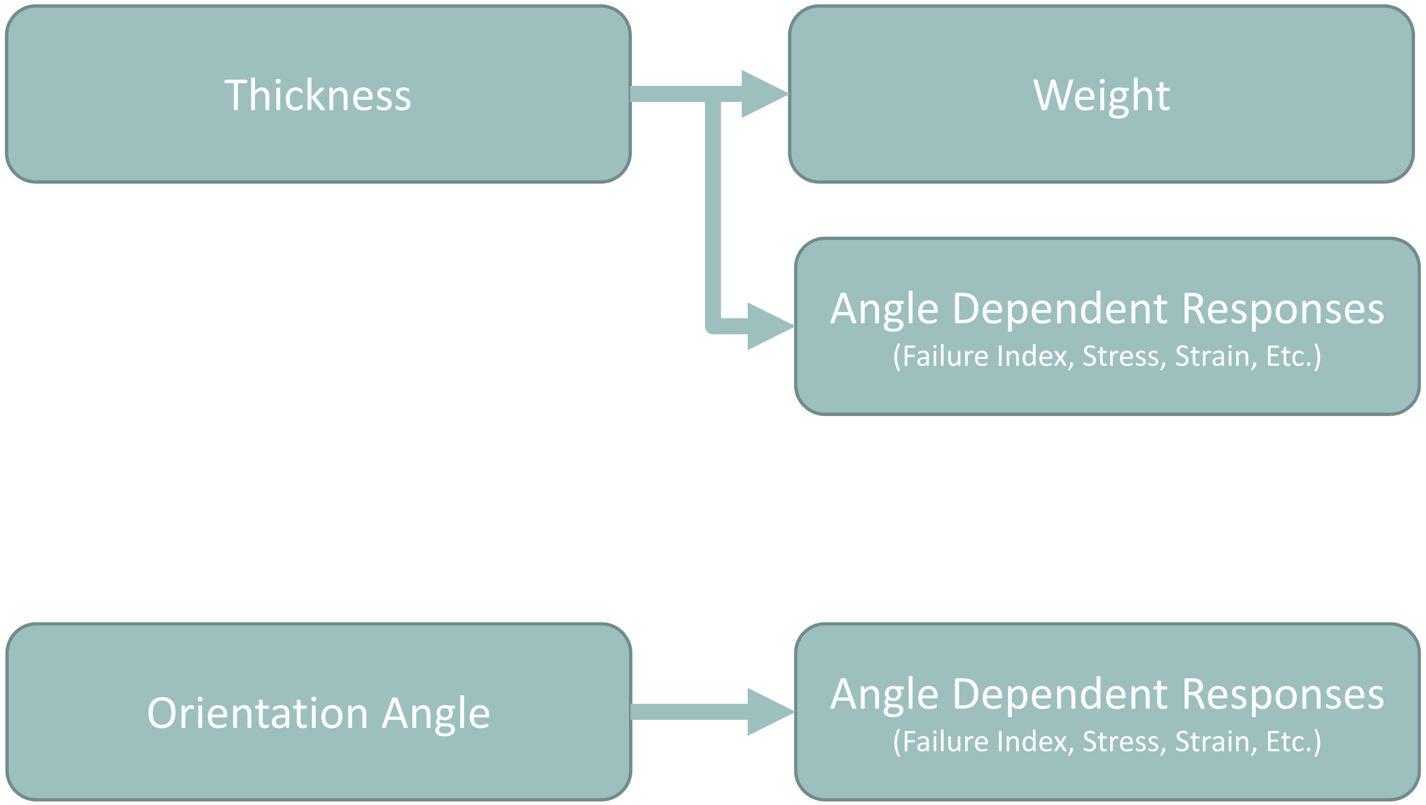


Why do the orientation angles not change?

Thickness influences both weight and angle dependent responses.

Orientation angles only influence angle dependent responses.

The optimizer cares most about the objective. The optimizer only takes action against the constraints when the constraints are violated. This is why you see the thickness change often, but sometimes see the orientation angles unchanged.



Why do the orientation angles not change?

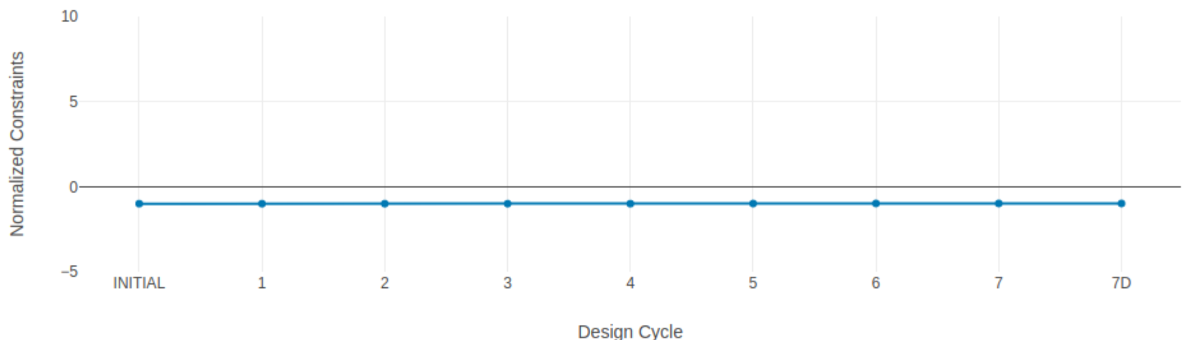
The reader may be tempted to state, "Intuition suggests the orientation should still change regardless."

In this example, the thickness variable starts at 1.0 and eventually ends at its lower bound of .01. Even when the thickness is at its thinnest, the normalized constraints remain negative. The moment when the normalized constraint is positive is never achieved, therefore the orientation angles are not changed.

If the normalized constraints are positive, i.e. violated constraints, the angles will change.

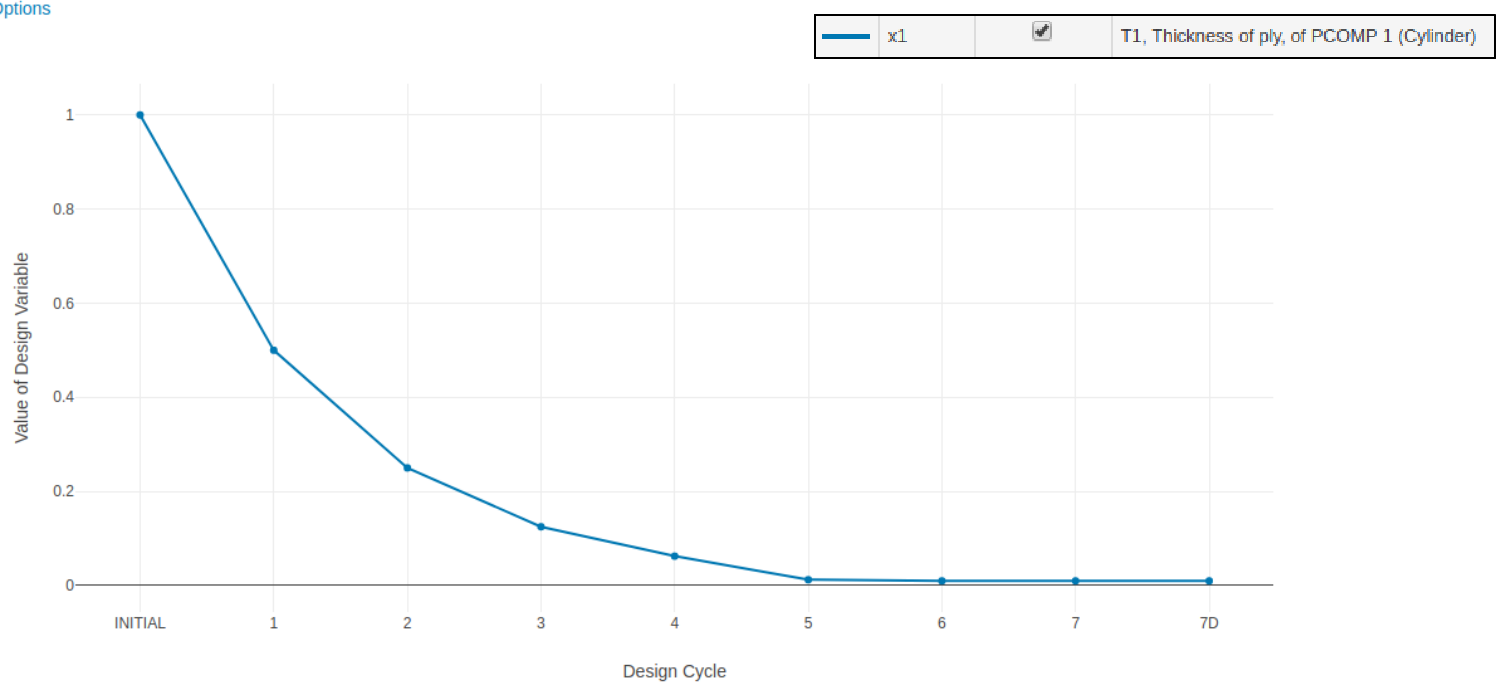
Normalized Constraints

+ Info



Design Variables

+ Options



Why do the orientation angles not change?

A change in orientation angles can be forced by reducing the lower bound of the thickness design variables.

During the optimization, the design can be become very thin and cause constraints to become violated, which cause a change in orientation angles.

Before:

	Label	Status	Property	Property Description	Entry	Entry ID	Initial Value	Lower Bound	Upper Bound	Allowed Values
				Thickness						
	x1		T1	Thickness of ply	PCOMP	1	1.0	.01	10.	Allowed discrete values, example: 1.5, 2.
	x2		T2	Thickness of ply	PCOMP	1	1.0	.01	Upper	Allowed discrete values, example: 1.5, 2.
	x3		T3	Thickness of ply	PCOMP	1	1.0	.01	Upper	Allowed discrete values, example: 1.5, 2.

After:

	Label	Status	Property	Property Description	Entry	Entry ID	Initial Value	Lower Bound	Upper Bound	Allowed Values
				Thickness						
	x1		T1	Thickness of ply	PCOMP	1	1.0	.0001	10.	Allowed discrete values, example: 1.5, 2.
	x2		T2	Thickness of ply	PCOMP	1	1.0	.0001	Upper	Allowed discrete values, example: 1.5, 2.
	x3		T3	Thickness of ply	PCOMP	1	1.0	.0001	Upper	Allowed discrete values, example: 1.5, 2.

Why do the orientation angles not change?

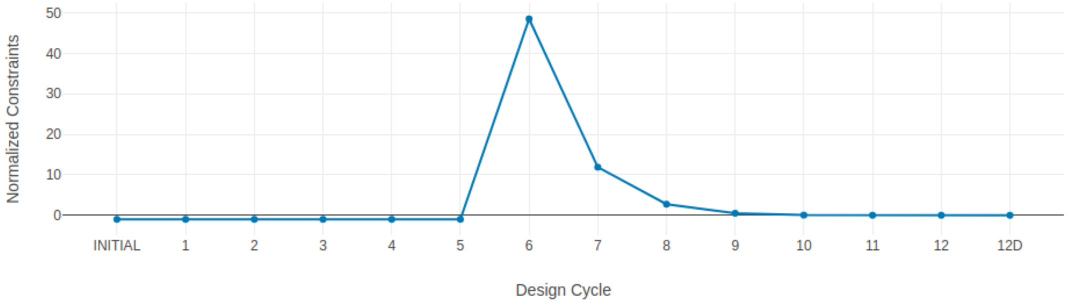
After reducing the lower bound on the thickness variables, the normalized constraint become positive, indicating a violated constraint, during design cycle 6. Note that the orientation angles also change in the same design cycle.

The reader may notice that after design cycle 7 and 8, the constraints are violated, but the orientation angles are now unchanging. At this point, the thickness is the variable being dominantly changed but is not shown in the variable plot.

To conclude, the orientation angles will change if an angle dependent constraint is violated.

Normalized Constraints

+ Info



Design Variables