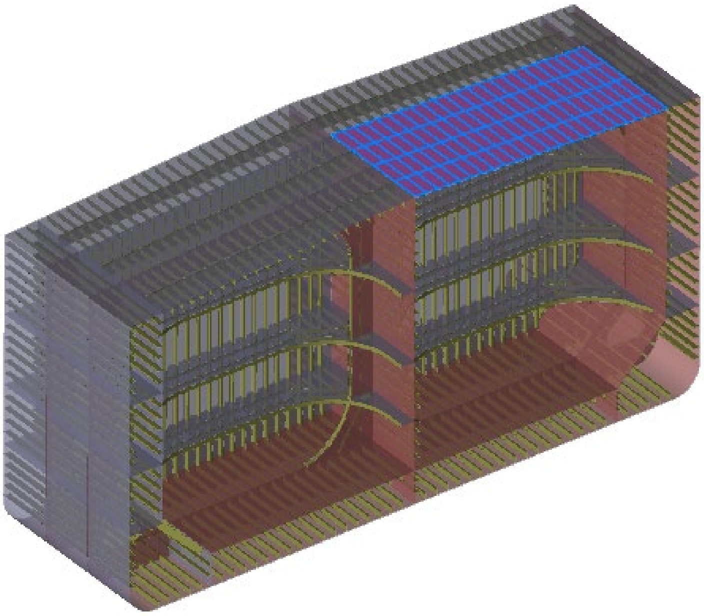
Sesam Tutorial

RP-C201 Buckling & Yield Check

by Sesam Core

DNV – Digital Solutions

Table of contents



This workshop will explain how to perform RP-C201 Section 6 - 6.2 SCM2 buckling check and DNV-OS-C101 element yield check using Sesam Core.

Below programs are used for this tutorial

- GeniE 8.12
- Sestra 10.20
- Xtract 6.4

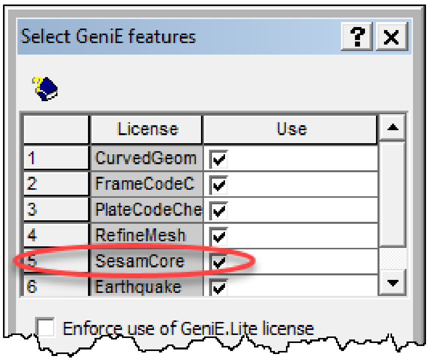
1 RUN ANALYSIS IN GENIE.

In this chapter, the structural model will be prepared and exported from GeniE for further use in Sesam Core.

Modelling is not in the scope of this tutorial so the structural model is provided. For more information about how to the model in GeniE please check the GeniE tutorials in the help section.

1.1 Open GeniE Model model into GeniE

- The required fatigue functionality in GeniE requires a **GENIE__SCORE** license as shown in the **Edit > License / features** dialog.
- The required fatigue functionality in GeniE requires a **GENIE__SCORE** license as shown in the Edit > License / features dialog.



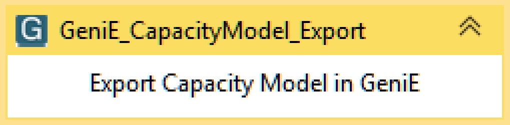
Select GeniE features

	License	Use
1	CurvedGeom	☑
2	FrameCodeC	☑
3	PlateCodeChe	☑
4	RefineMesh	☑
5	SesamCore	☑
6	Earthquake	☑

□ Enforce use of GeniE.Lite license

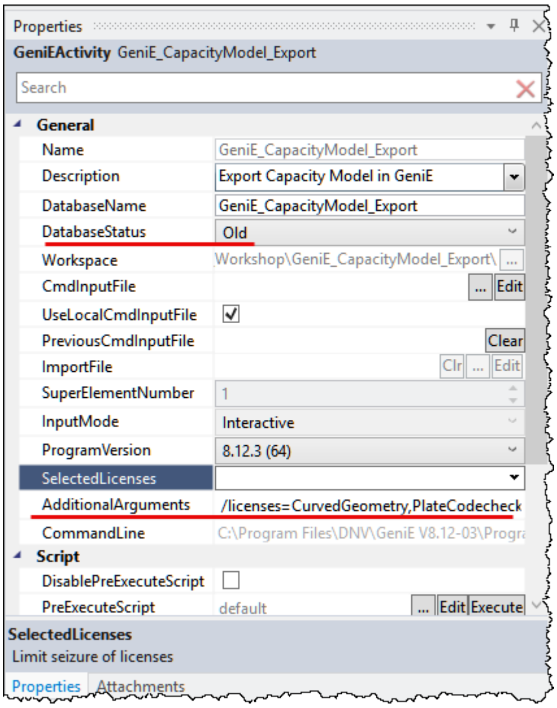
- Above options will be activated by defining them through arguments.

- Right click the **GeniE_CapacityModel_Export** activity in Sesam Manager and click **Run**



GeniE_CapacityModel_Export

Export Capacity Model in GeniE



Properties

GeniEActivity GeniE_CapacityModel_Export

Search

General

Name	GeniE_CapacityModel_Export
Description	Export Capacity Model in GeniE
DatabaseName	GeniE_CapacityModel_Export
DatabaseStatus	Old
Workspace	Workshop\GeniE_CapacityModel_Export\
CmdInputFile	
UseLocalCmdInputFile	☑
PreviousCmdInputFile	
ImportFile	
SuperElementNumber	1
InputMode	Interactive
ProgramVersion	8.12.3 (64)
SelectedLicenses	
AdditionalArguments	/licenses=CurvedGeometry,PlateCodecheck
CommandLine	C:\Program Files\DNV\GeniE V8.12-03\Progra

Script

DisablePreExecuteScript	□
PreExecuteScript	default

SelectedLicenses

Limit seizure of licenses

- DatabaseStatus : Old
- AdditionalArguments: **/licenses=CurvedGeometry,PlateCodecheck,RefineMesh,SesamCore /mode=full**
  - Enable SesamCore license

Then GeniE will open the database, and the model will be displayed

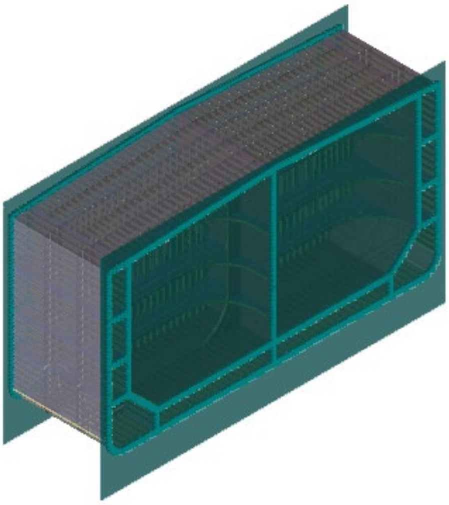


Figure 1 Global model of substructure

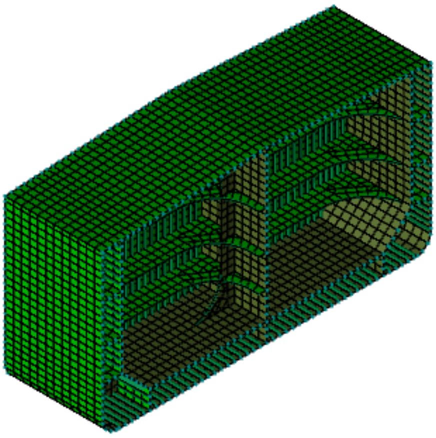
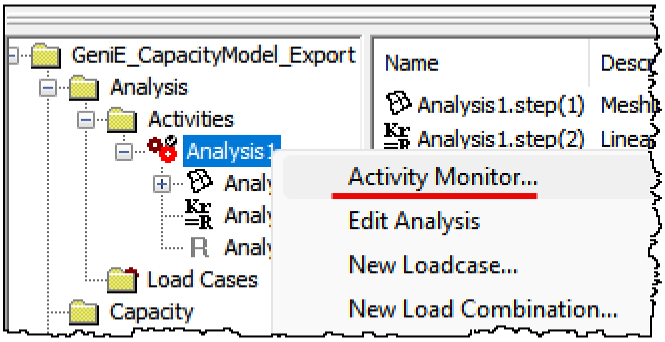


Figure 2 Mesh of global model

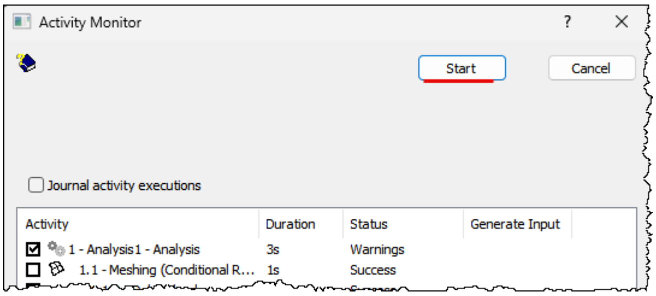
1.2 Run Analysis

Run analysis in GeniE.

- Open Activity Monitor



- Start the analysis



□ Journal activity executions

Activity	Duration	Status	Generate Input
☑ 1 - Analysis1 - Analysis	3s	Warnings	
□ 1.1 - Meshing (Conditional R...	1s	Success	

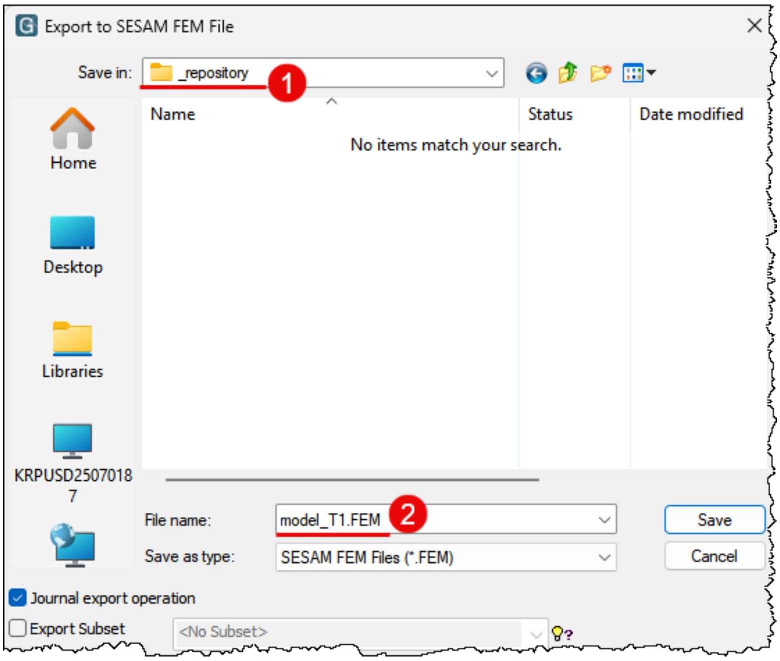
Then analysis will be executed, and result file will be generated.

2 EXPORT MODEL & RESULT IN GENIE.

2.1 Export Model

Export model into **_repository** folder to be used with SesamCore.

- Select the **Mesh – All** in the GeniE toolbar.
- **File | Export | FEM file…**
- Save the model file
    - folder : **_repository** (located one level above from current GeniE folder)
    - File name : **model_T1.FEM**
    -



Export to SESAM FEM File

Save in: _repository 1

File name: model_T1.FEM 2

Save as type: SESAM FEM Files (*.FEM)

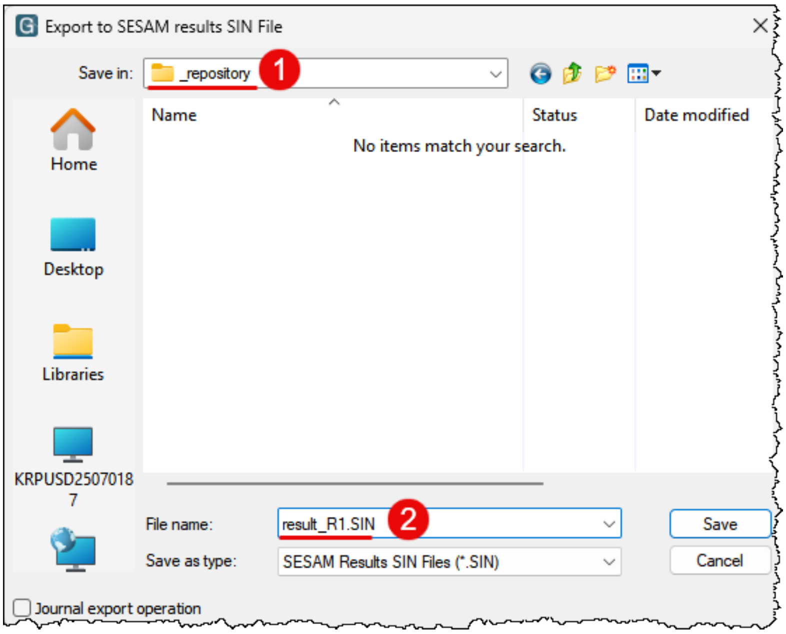
☑ Journal export operation

□ Export Subset <No Subset>

2.2 Export Result File

Export model into **_repository** folder to be used with SesamCore.

- **File | Export | Result SIN file…**
- Save the result file
    - folder : **_repository** (located one level above from current GeniE folder)
    - File name : **result_T1.SIN**



Export to SESAM results SIN File

Save in: _repository 1

File name: result_R1.SIN 2

Save as type: SESAM Results SIN Files (*.SIN)

□ Journal export operation

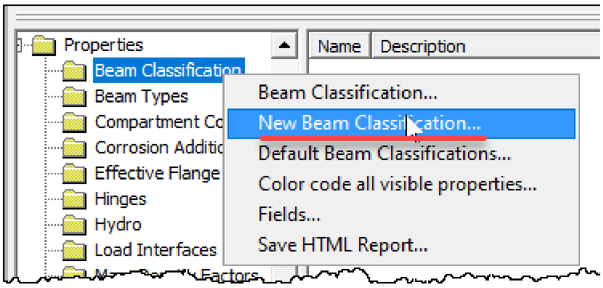
❖ Note.
Sesam Core can be run with Sestra, so running Sestra before Sesam Core is not mandatory.

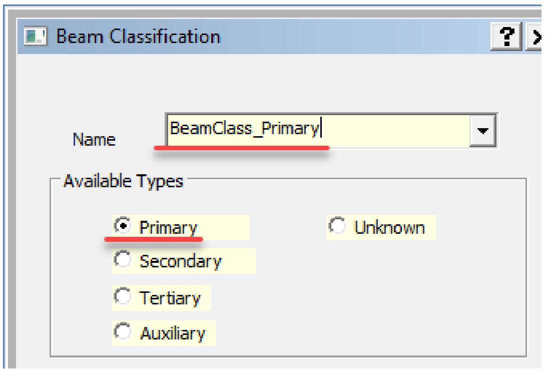
3 EXPORT CAPACITY MODEL IN GENIE

3.1 Define Primary Girders as Primary Classification.

Define primary girders as primary classification.

- Create **new Beam Classification**





Name BeamClass_Primary

Available Types

● Primary ○ Unknown

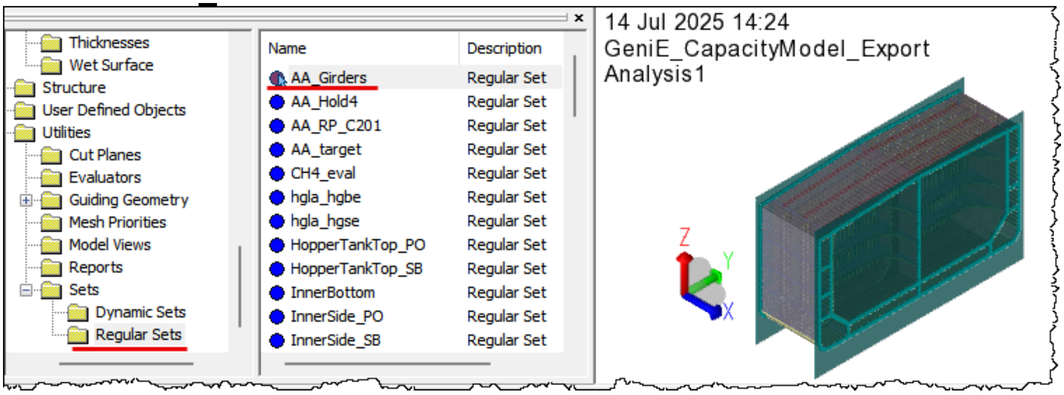
○ Secondary

○ Tertiary

○ Auxiliary

- Name : **BeamClass_Primary**
- Type : **Primary**

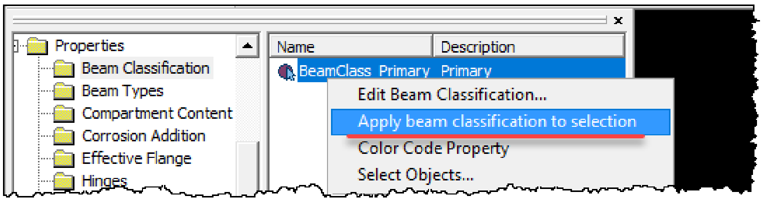
- Select set **AA_Girders**



Name	Description
AA_Girders	Regular Set
AA_Hold4	Regular Set
AA_RP_C201	Regular Set
AA_target	Regular Set
CH4_eval	Regular Set
hgla_hgbe	Regular Set
hgla_hgse	Regular Set
HopperTankTop_PO	Regular Set
HopperTankTop_SB	Regular Set
InnerBottom	Regular Set
InnerSide_PO	Regular Set
InnerSide_SB	Regular Set

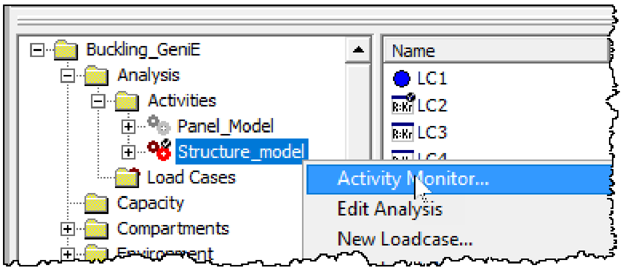
14 Jul 2025 14:24
GeniE_CapacityModel_Export
Analysis1

- **Apply BeamClassification** property

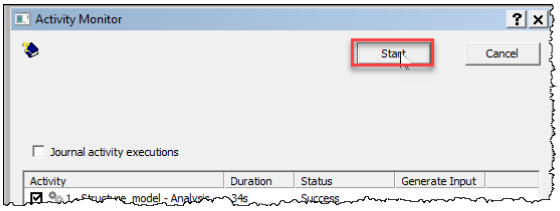


Run activity monitor, to update the mesh.

- **Open Activity Monitor** for “**Structure Model**”



- Then “**Start**” the activity

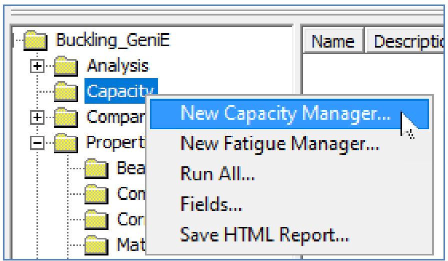


□ Journal activity executions

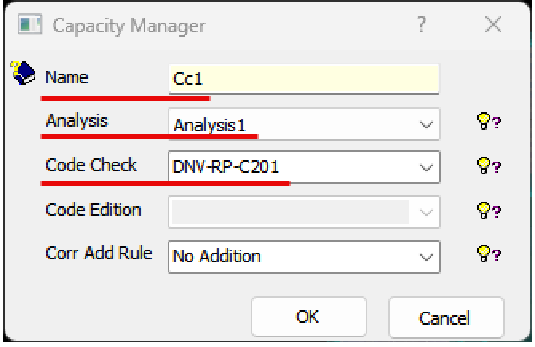
Activity	Duration	Status	Generate Input
☑ Structure_model - Analysis	34s	Success	

3.2 Defining the capacity manager

- Click **right mouse button (RMB)** on the **Capacity** folder to create a new capacity manager.



- Select the **Analysis1** analysis, **DNV-RP-C201** code check and no corrosion addition rule.



Name: Cc1

Analysis: Analysis1

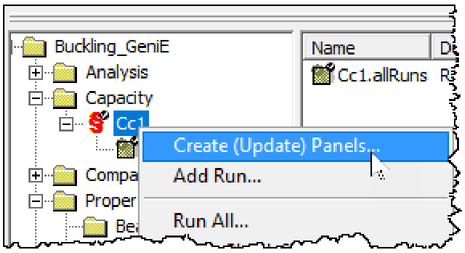
Code Check: DNV-RP-C201

Code Edition:

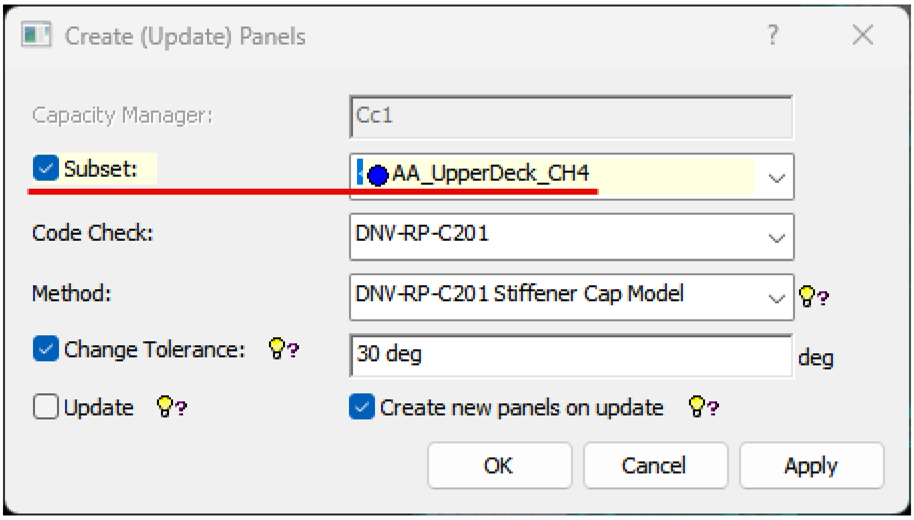
Corr Add Rule: No Addition

3.3 Creating the panels

- **Right mouse button (RMB)** on the newly created **capacity model** and select to create panels.



- On the new menu select the subset **Column1_BHD** and keep the other options as default, then press OK.



Capacity Manager: Cc1

☑ Subset: AA_UpperDeck_CH4

Code Check: DNV-RP-C201

Method: DNV-RP-C201 Stiffener Cap Model

☑ Change Tolerance: 30 deg

□ Update

☑ Create new panels on update

The panels will be created based on the selected subset using the stiffener capacity model (SCM2).

- To view the capacity model change the view type to **Capacity Models**

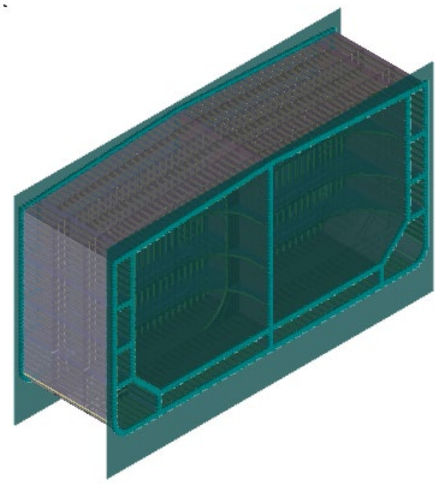


Figure - Model of **Column1_BHD**

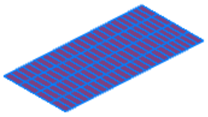
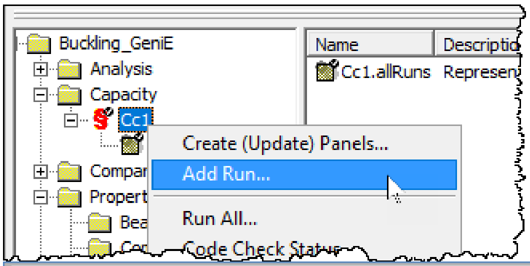


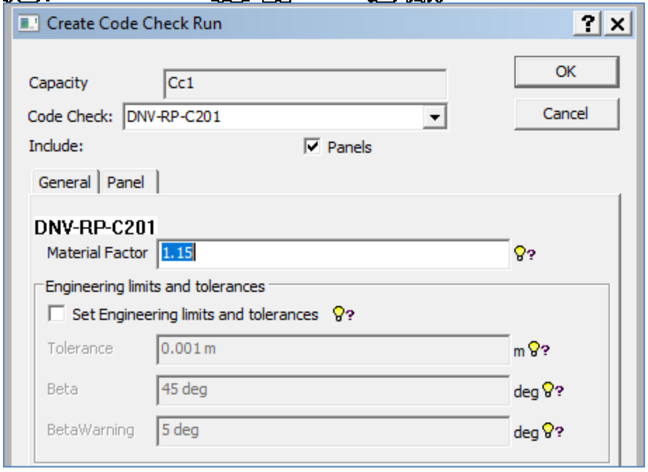
Figure - Panels in the capacity model

3.4 Create the code check run

- Press **Right mouse button (RMB)** on the capacity model and select **Add run** to create the code check run



- Confirm that **DNV-RP-C-201** is selected for the codecheck option.
- In the General tab the material factor can be adjusted for other limit states. For ULS the default value of **1.15** is adequate.
- Press OK to close the dialog.

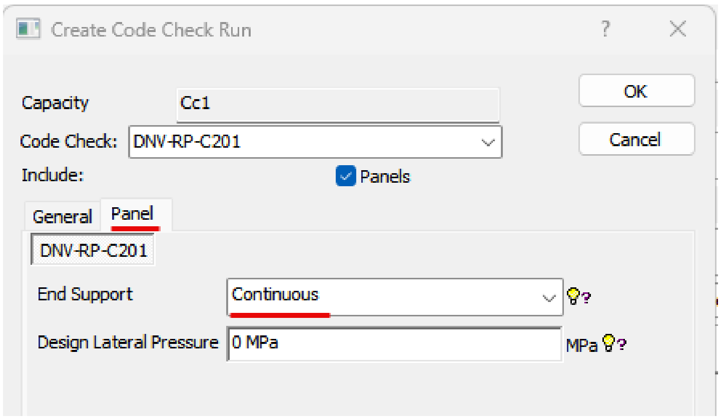


Capacity: Cc1
Code Check: DNV-RP-C201
Include: ☑ Panels

DNV-RP-C201
Material Factor: 1.15

Engineering limits and tolerances
☐ Set Engineering limits and tolerances
Tolerance: 0.001 m
Beta: 45 deg
BetaWarning: 5 deg

- In the **Panel** tab the end support for the stiffeners can be set to **continuous** or **snipped**. This is applied to both ends of the stiffeners. For this case it is ok to leave it as continuous



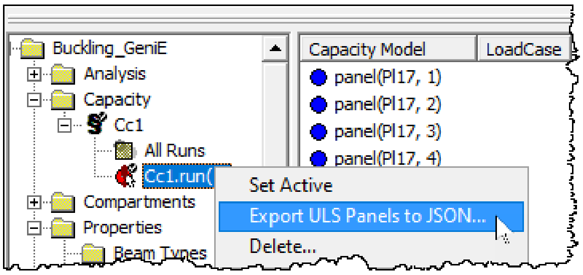
Capacity: Cc1
Code Check: DNV-RP-C201
Include: ☑ Panels

DNV-RP-C201
End Support: Continuous
Design Lateral Pressure: 0 MPa

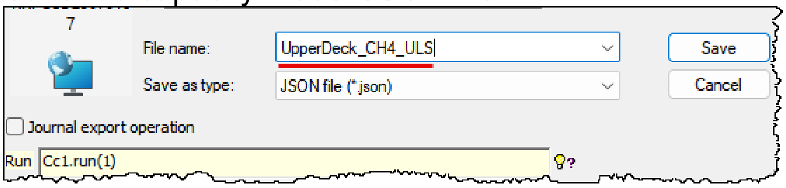
3.5 Export capacity model

Export the capacity model to a JSON file to be used with SesamCore.

- RMB click the code check run and and select **Export ULS Panels to JSON**



- Save the capacity model JSON file


File name: UpperDeck_CH4_ULS

Save as type: JSON file (*.json)

□ Journal export operation

Run Cc1.run(1)

  - folder : **_repository** (located one level above from current GeniE folder)
  - File name : **UpperDeck_CH4_ULS.json**

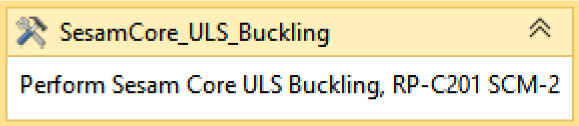
The JSON file will be created in the GeniE workspace folder. This file and the structural model file will be used in Sesam Core to run the ULS code check.

Close GeniE.

4 PERFORM BUCKLING CHECK IN SESAM CORE

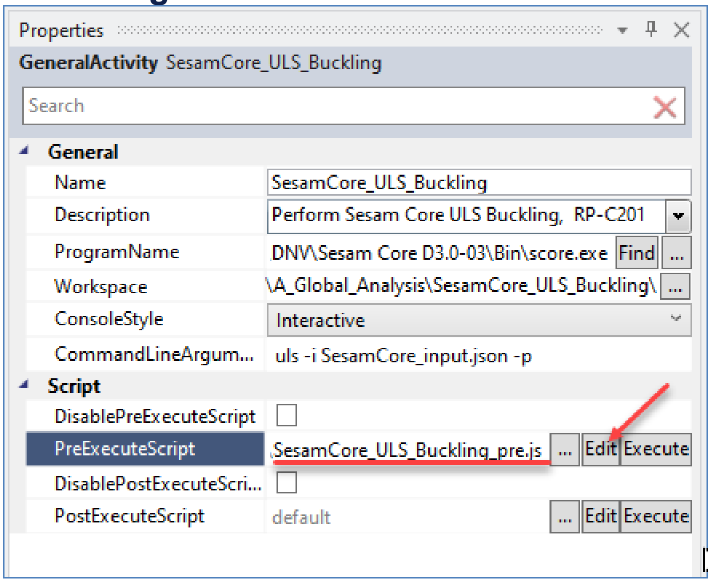
This chapter explains how to perform **buckling** check. As a result of Sestra run, there will be result R#.SIN file, which will be used to calculate buckling in **Sesam Core**, calculates the ultimate limit state according to **DNV-RP-C201**.

4.1 Check "PreExecuteScript" before running Sesam Core.


SesamCore_ULS_Buckling

Perform Sesam Core ULS Buckling, RP-C201 SCM-2

- Click "**Edit**" to check the script.



GeneralActivity SesamCore_ULS_Buckling

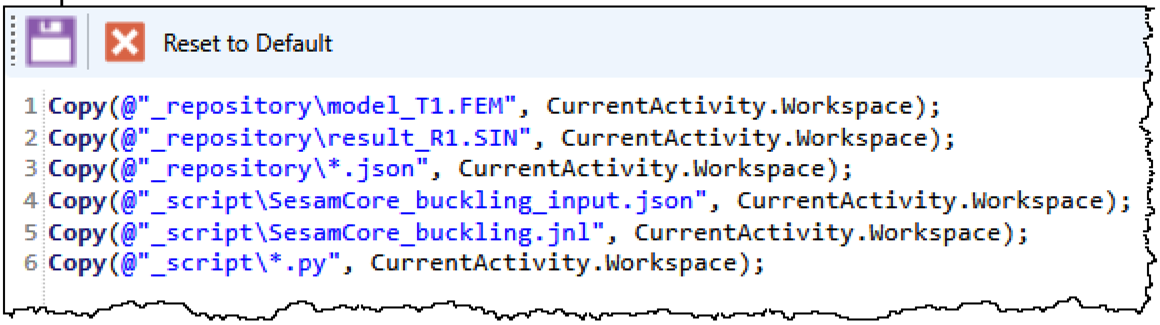
General

Name	SesamCore_ULS_Buckling
Description	Perform Sesam Core ULS Buckling, RP-C201
ProgramName	DNV\Sesam Core D3.0-03\Bin\score.exe
Workspace	\A_Global_Analysis\SesamCore_ULS_Buckling\
ConsoleStyle	Interactive
CommandLineArgum...	uls -i SesamCore_input.json -p

Script

DisablePreExecuteScript	□
PreExecuteScript	SesamCore_ULS_Buckling_pre.js
DisablePostExecuteScri...	□
PostExecuteScript	default

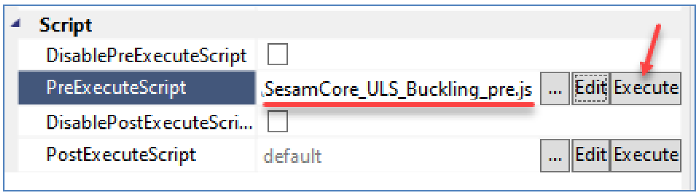
Script is as below



```
Copy(@"_repository\model_T1.FEM", CurrentActivity.Workspace);
Copy(@"_repository\result_R1.SIN", CurrentActivity.Workspace);
Copy(@"_repository\*.json", CurrentActivity.Workspace);
Copy(@"_script\SesamCore_buckling_input.json", CurrentActivity.Workspace);
Copy(@"_script\SesamCore_buckling.jnl", CurrentActivity.Workspace);
Copy(@"_script\*.py", CurrentActivity.Workspace);
```

- Copy below files from _repository folder to Sesam Core folder
  - **model_T3.FEM** : Structure Model
  - **result_R3.SIN** : Sestra result file
  - **UpperDeck_CH4_ULS.json** : Capacity Model data for ULS
- Copy below files from _script to Sesam Core folder
  - **SesamCore_buckling_input.json** : Sesam Core Input File
  - **SesamCore_buckling.jnl** : Sesam Core Journal File
  - (Optional) **Read_SesamCore_ULS_GeniE_Uf_Plot_v02.py** : Python example file for post-processing

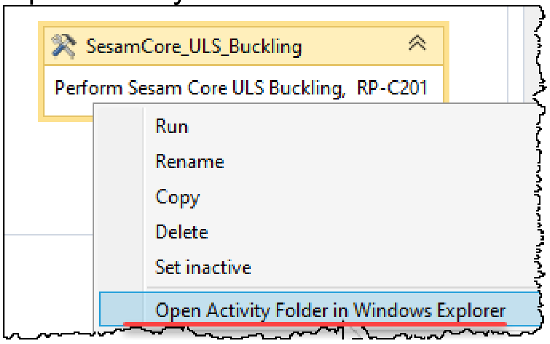
- Click “**Execute**” to run that script.



Script	
DisablePreExecuteScript	□
PreExecuteScript	SesamCore_ULS_Buckling_pre.js ... Edit Execute
DisablePostExecuteScri...	□
PostExecuteScript	default ... Edit Execute

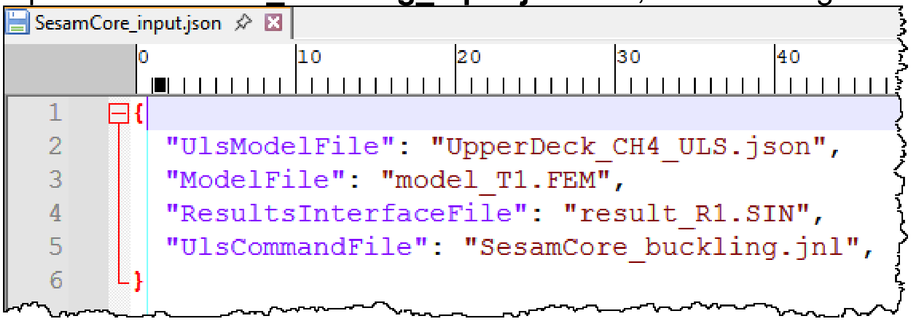
4.2 Check input files

- Open activity folder of Sesam Core



SesamCore_ULS_Buckling
Perform Sesam Core ULS Buckling, RP-C201
Run
Rename
Copy
Delete
Set inactive
Open Activity Folder in Windows Explorer

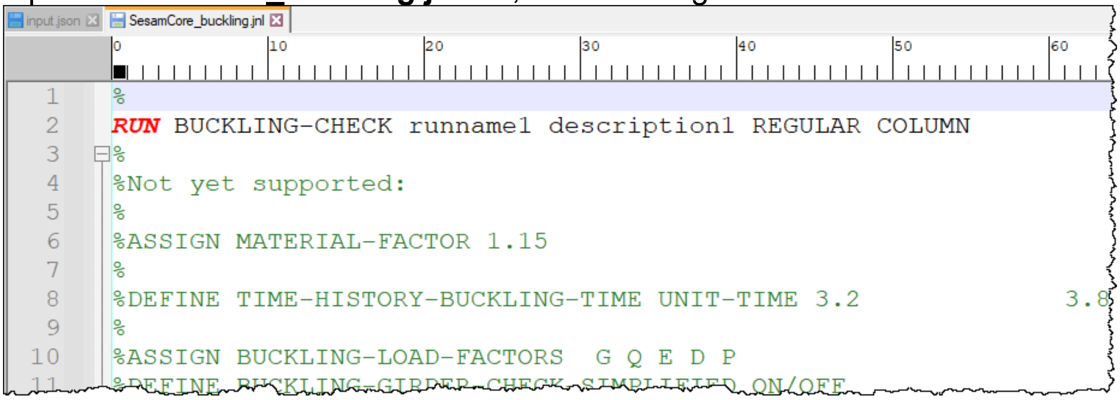
- Open **SesamCore_buckling_input.json** file, then investigate it.



```
{
  "UlsModelFile": "UpperDeck_CH4_ULS.json",
  "ModelFile": "model_T1.FEM",
  "ResultsInterfaceFile": "result_R1.SIN",
  "UlsCommandFile": "SesamCore_buckling.jnl",
}
```

  - In this input file, ULS model file, FEM model file, result file, and Sesam Core journal file are specified.

- Open **SesamCore_buckling.jnl** file, then investigate it.

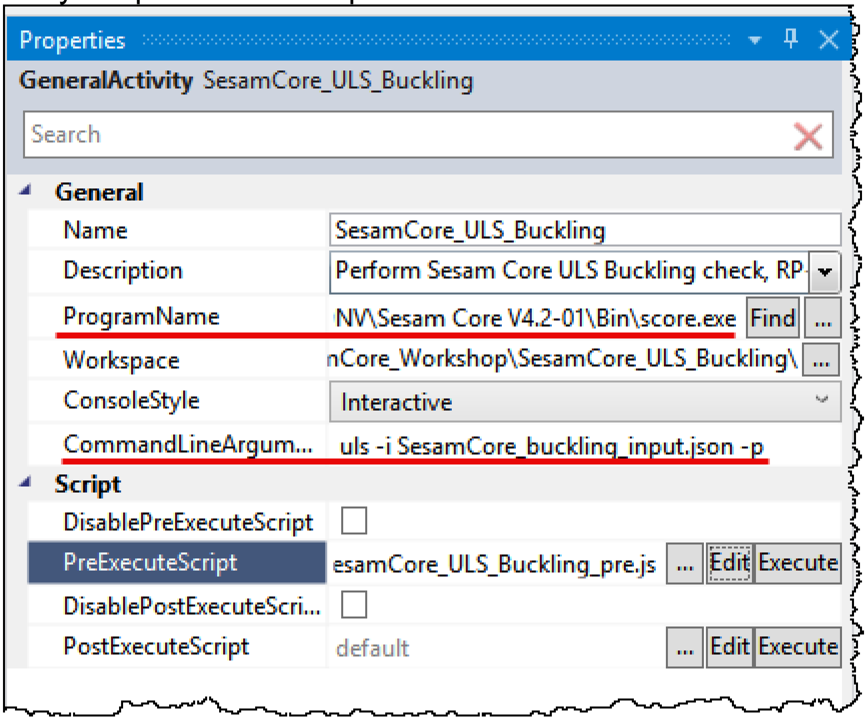


```
%
RUN BUCKLING-CHECK runname1 description1 REGULAR COLUMN
%
%Not yet supported:
%
%ASSIGN MATERIAL-FACTOR 1.15
%
%DEFINE TIME-HISTORY-BUCKLING-TIME UNIT-TIME 3.2            3.8
%
%ASSIGN BUCKLING-LOAD-FACTORS  G Q E D P
%DEFINE BUCKLING-GIRDER-CHECK-SIMPLIFIED ON/OFF
```

  - Here only “**RUN BUCKLING-CHECK**” command is used. But there are also some example commands, which can be used un Sesam Core ULS check.
  - For more details, please refer Sesam Core User manual.

4.3 Execute Sesam Core

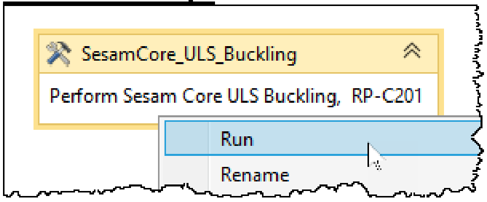
- To run Sesam Core, path of the Sesam Core and command line arguments should be defined.
  They're specified in "Properties" as below.



GeneralActivity SesamCore_ULS_Buckling

General	
Name	SesamCore_ULS_Buckling
Description	Perform Sesam Core ULS Buckling check, RP-
ProgramName	NV\Sesam Core V4.2-01\Bin\score.exe
Workspace	nCore_Workshop\SesamCore_ULS_Buckling\
ConsoleStyle	Interactive
CommandLineArgum...	uls -i SesamCore_buckling_input.json -p
Script	
DisablePreExecuteScript	□
PreExecuteScript	esamCore_ULS_Buckling_pre.js
DisablePostExecuteScri...	□
PostExecuteScript	default

- Then program will be executed as below.
  - `Command: "C:\Program Files\DNV\Sesam Core V4.2-01\Bin\score.exe" uls -i SesamCore_buckling_input.json -p`
  - Here "`ULS`" argument is to perform ULS buckling check.
  - "`-i SesamCore_buckling_input.json`" specifying the input file of Sesam Core
  - "`-p`" is to perform Sesam Core Post-processing only. Without that command, Sestra will be executed simultaneously.
- Correct path of the "score.exe", if want to use different version of Sesam Core
- **Run this step** to execute Seasm Core.



SesamCore_ULS_Buckling

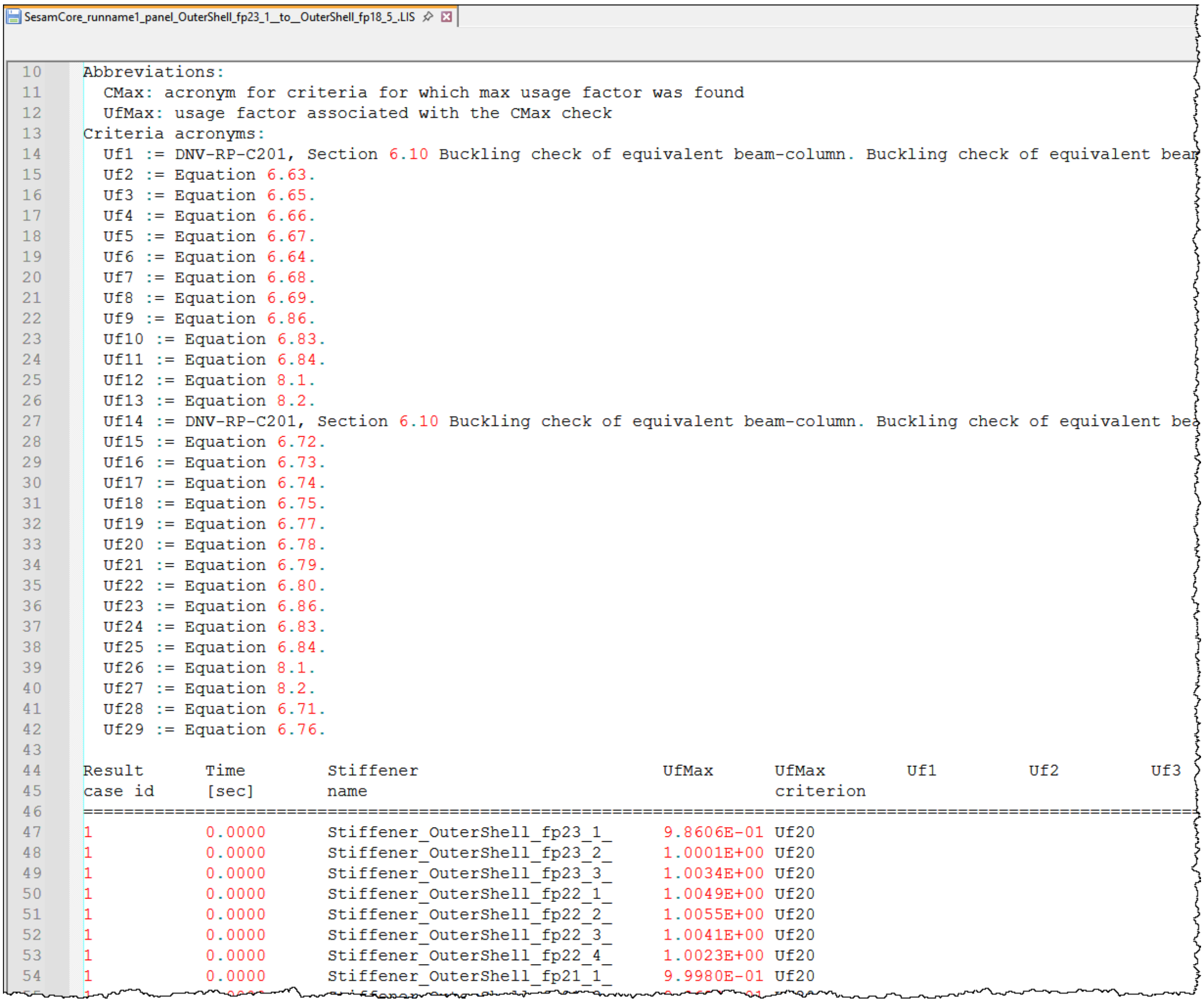
Perform Sesam Core ULS Buckling, RP-C201

- If user want to run in command line (CLI), then use run_SesamCore_ULS_buckling.bat batch file in the activity folder.

4.4 Inspect the results #.lis files

After finishing Sesam Core ULS, user may check result and feed-back files

- **SCORE.MLG** : feed-back from the Sesam Core
- **#.csv / #.lis files** : buckling check results for each panel.



SesamCore_runname1_panel_OuterShell_fp23_1_to_OuterShell_fp18_5_.LIS

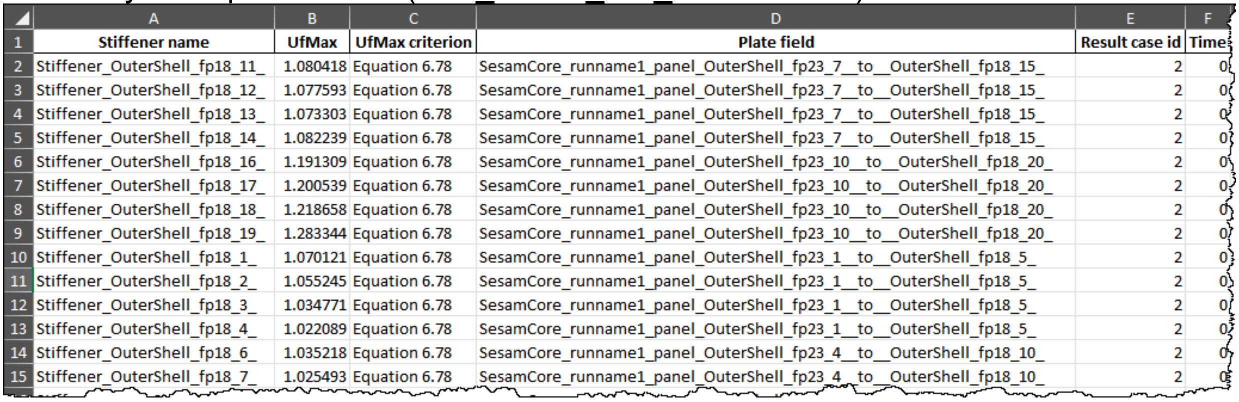
```
Abbreviations:
  CMax: acronym for criteria for which max usage factor was found
  UfMax: usage factor associated with the CMax check
Criteria acronyms:
  Uf1  := DNV-RP-C201, Section 6.10 Buckling check of equivalent beam-column. Buckling check of equivalent bea
  Uf2  := Equation 6.63.
  Uf3  := Equation 6.65.
  Uf4  := Equation 6.66.
  Uf5  := Equation 6.67.
  Uf6  := Equation 6.64.
  Uf7  := Equation 6.68.
  Uf8  := Equation 6.69.
  Uf9  := Equation 6.86.
  Uf10  := Equation 6.83.
  Uf11  := Equation 6.84.
  Uf12  := Equation 8.1.
  Uf13  := Equation 8.2.
  Uf14  := DNV-RP-C201, Section 6.10 Buckling check of equivalent beam-column. Buckling check of equivalent be
  Uf15  := Equation 6.72.
  Uf16  := Equation 6.73.
  Uf17  := Equation 6.74.
  Uf18  := Equation 6.75.
  Uf19  := Equation 6.77.
  Uf20  := Equation 6.78.
  Uf21  := Equation 6.79.
  Uf22  := Equation 6.80.
  Uf23  := Equation 6.86.
  Uf24  := Equation 6.83.
  Uf25  := Equation 6.84.
  Uf26  := Equation 8.1.
  Uf27  := Equation 8.2.
  Uf28  := Equation 6.71.
  Uf29  := Equation 6.76.

Result      Time        Stiffener                        UfMax      UfMax        Uf1         Uf2         Uf3
case id     [sec]       name                                        criterion
==============================================================================================================
1           0.0000      Stiffener_OuterShell_fp23_1_     9.8606E-01 Uf20
1           0.0000      Stiffener_OuterShell_fp23_2_     1.0001E+00 Uf20
1           0.0000      Stiffener_OuterShell_fp23_3_     1.0034E+00 Uf20
1           0.0000      Stiffener_OuterShell_fp22_1_     1.0049E+00 Uf20
1           0.0000      Stiffener_OuterShell_fp22_2_     1.0055E+00 Uf20
1           0.0000      Stiffener_OuterShell_fp22_3_     1.0041E+00 Uf20
1           0.0000      Stiffener_OuterShell_fp22_4_     1.0023E+00 Uf20
1           0.0000      Stiffener_OuterShell_fp21_1_     9.9980E-01 Uf20
```

4.5 (Optional) Make summary table and plots for each panel

Now the results are saved multiple piles per each panel. To get summary table for max usage factor, user may use Python script, until Sesam Core supports it.

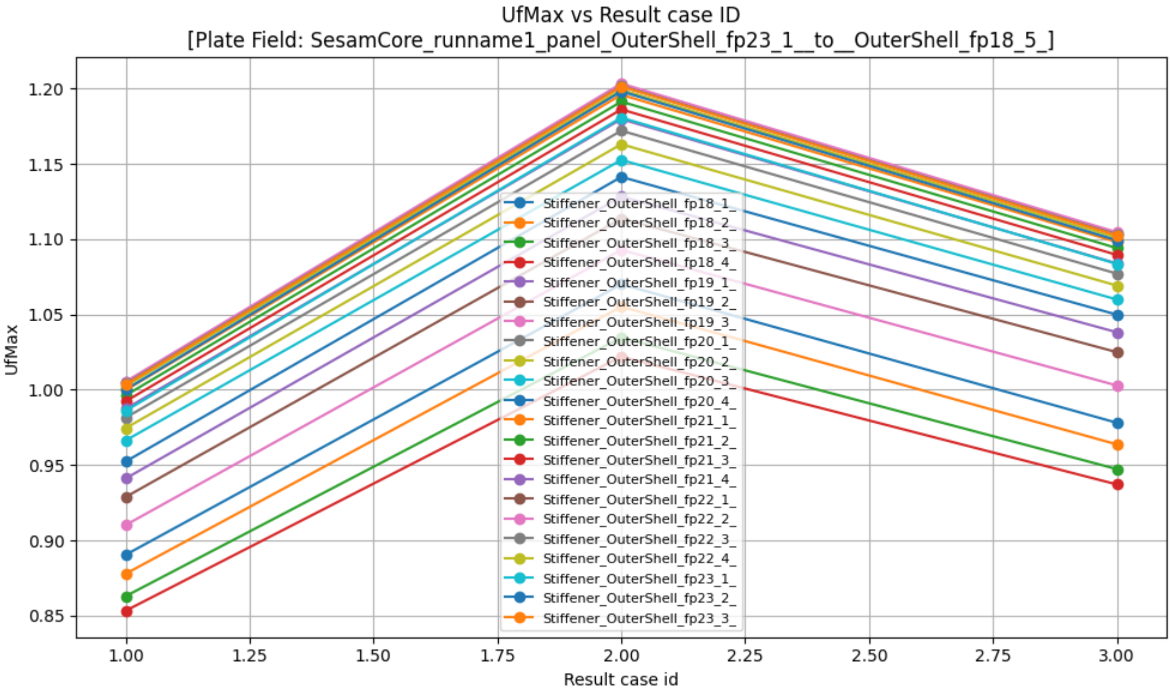
There is one example Python script file, “Read_SesamCore_ULS_GeniE_Uf_Plot_v02.py”. Then run it in the Sesam Core run folder. Then user will get summary table and Uf plots for each panel.

- Summary table per stiffener (Max_UfMax_Per_Stiffener.xlsx)



	A	B	C	D	E	F
1	Stiffener name	UfMax	UfMax criterion	Plate field	Result case id	Time
2	Stiffener_OuterShell_fp18_11_	1.080418	Equation 6.78	SesamCore_runname1_panel_OuterShell_fp23_7__to__OuterShell_fp18_15_	2	0
3	Stiffener_OuterShell_fp18_12_	1.077593	Equation 6.78	SesamCore_runname1_panel_OuterShell_fp23_7__to__OuterShell_fp18_15_	2	0
4	Stiffener_OuterShell_fp18_13_	1.073303	Equation 6.78	SesamCore_runname1_panel_OuterShell_fp23_7__to__OuterShell_fp18_15_	2	0
5	Stiffener_OuterShell_fp18_14_	1.082239	Equation 6.78	SesamCore_runname1_panel_OuterShell_fp23_7__to__OuterShell_fp18_15_	2	0
6	Stiffener_OuterShell_fp18_16_	1.191309	Equation 6.78	SesamCore_runname1_panel_OuterShell_fp23_10__to__OuterShell_fp18_20_	2	0
7	Stiffener_OuterShell_fp18_17_	1.200539	Equation 6.78	SesamCore_runname1_panel_OuterShell_fp23_10__to__OuterShell_fp18_20_	2	0
8	Stiffener_OuterShell_fp18_18_	1.218658	Equation 6.78	SesamCore_runname1_panel_OuterShell_fp23_10__to__OuterShell_fp18_20_	2	0
9	Stiffener_OuterShell_fp18_19_	1.283344	Equation 6.78	SesamCore_runname1_panel_OuterShell_fp23_10__to__OuterShell_fp18_20_	2	0
10	Stiffener_OuterShell_fp18_1_	1.070121	Equation 6.78	SesamCore_runname1_panel_OuterShell_fp23_1__to__OuterShell_fp18_5_	2	0
11	Stiffener_OuterShell_fp18_2_	1.055245	Equation 6.78	SesamCore_runname1_panel_OuterShell_fp23_1__to__OuterShell_fp18_5_	2	0
12	Stiffener_OuterShell_fp18_3_	1.034771	Equation 6.78	SesamCore_runname1_panel_OuterShell_fp23_1__to__OuterShell_fp18_5_	2	0
13	Stiffener_OuterShell_fp18_4_	1.022089	Equation 6.78	SesamCore_runname1_panel_OuterShell_fp23_1__to__OuterShell_fp18_5_	2	0
14	Stiffener_OuterShell_fp18_6_	1.035218	Equation 6.78	SesamCore_runname1_panel_OuterShell_fp23_4__to__OuterShell_fp18_10_	2	0
15	Stiffener_OuterShell_fp18_7_	1.025493	Equation 6.78	SesamCore_runname1_panel_OuterShell_fp23_4__to__OuterShell_fp18_10_	2	0

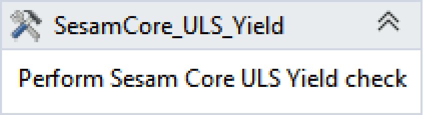
- Plots per plate field



5 PERFORM YIELD CHECK IN SESAM CORE

This chapter explains how to perform **yield check**. This step will be quite similar to buckling check, but will not require capacity model #.json file.

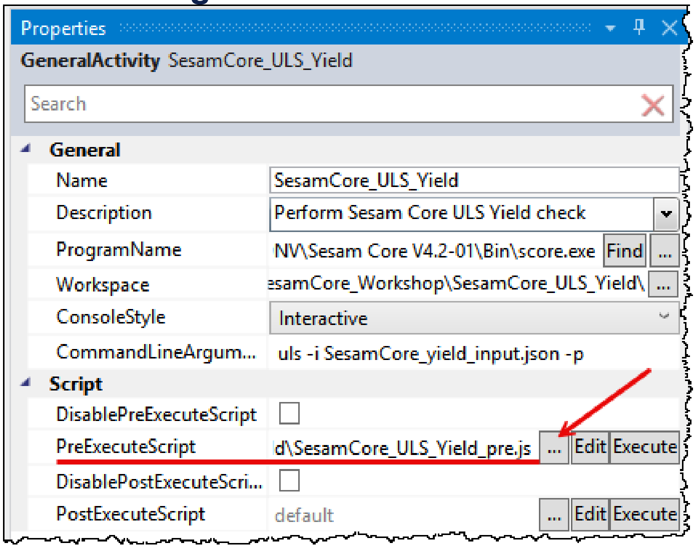
5.1 Check “PreExecuteScript” before running Sesam Core.



SesamCore_ULS_Yield

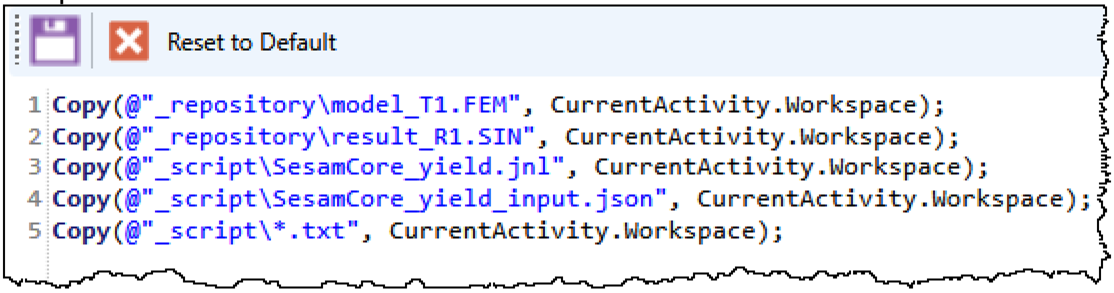
Perform Sesam Core ULS Yield check

- Click “**Edit**” to check the script.



Name: SesamCore_ULS_Yield
Description: Perform Sesam Core ULS Yield check
ProgramName: NV\Sesam Core V4.2-01\Bin\score.exe
Workspace: esamCore_Workshop\SesamCore_ULS_Yield\
ConsoleStyle: Interactive
CommandLineArgum...: uls -i SesamCore_yield_input.json -p
DisablePreExecuteScript: □
PreExecuteScript: d\SesamCore_ULS_Yield_pre.js
DisablePostExecuteScri...: □
PostExecuteScript: default

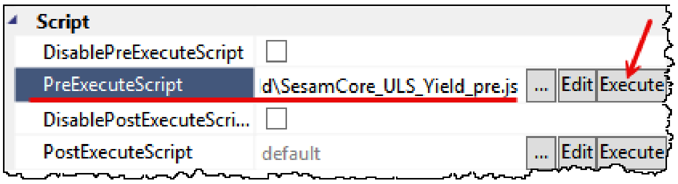
Script is as below



```
Copy(@"_repository\model_T1.FEM", CurrentActivity.Workspace);
Copy(@"_repository\result_R1.SIN", CurrentActivity.Workspace);
Copy(@"_script\SesamCore_yield.jnl", CurrentActivity.Workspace);
Copy(@"_script\SesamCore_yield_input.json", CurrentActivity.Workspace);
Copy(@"_script\*.txt", CurrentActivity.Workspace);
```

- Copy below files from _repository folder to Sesam Core folder
  - **model_T3.FEM** : Structure Model
  - **result_R3.SIN** : Sestra result file
- Copy below files from _script to Sesam Core folder
  - **SesamCore_yield_input.json** : Sesam Core Input File
  - **SesamCore_yield.jnl** : Sesam Core Journal File

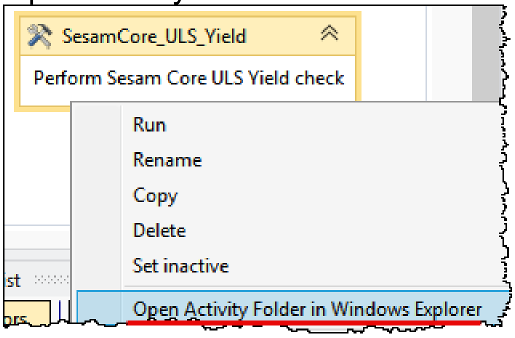
- Click “**Execute**” to run that script.



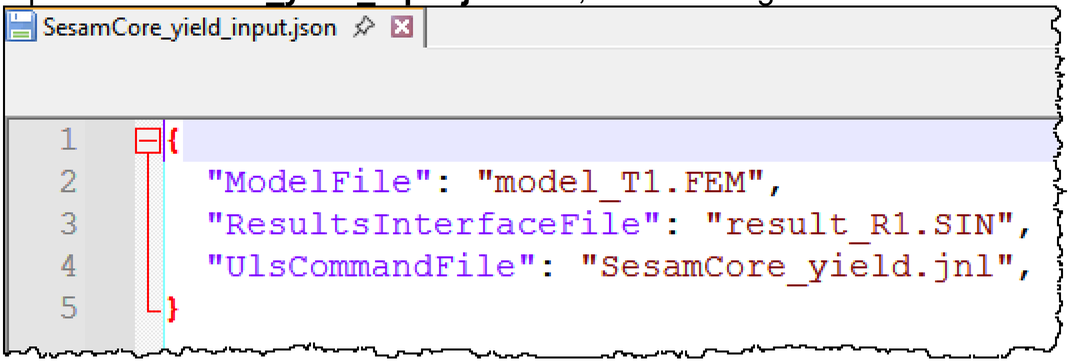
DisablePreExecuteScript: □
PreExecuteScript: d\SesamCore_ULS_Yield_pre.js
DisablePostExecuteScri...: □
PostExecuteScript: default

5.2 Check input files

- Open activity folder of Sesam Core


SesamCore_ULS_Yield
Perform Sesam Core ULS Yield check
Run
Rename
Copy
Delete
Set inactive
Open Activity Folder in Windows Explorer

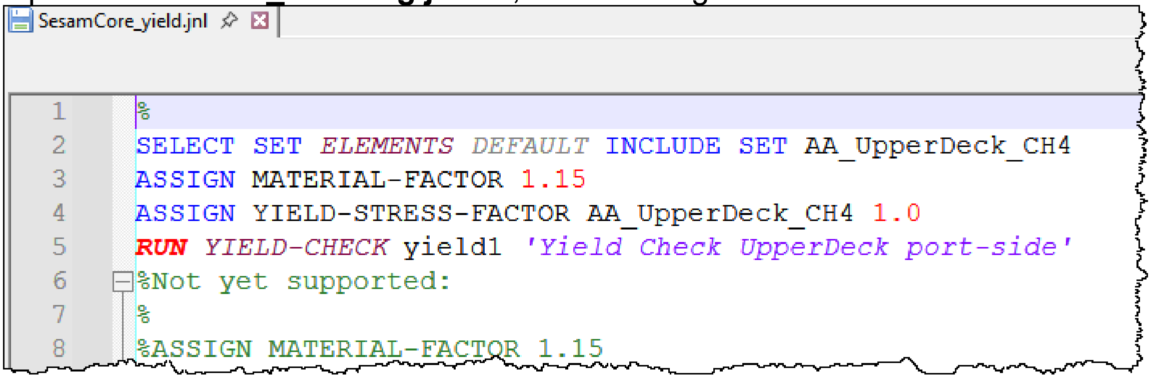
- Open **SesamCore_yield_input.json** file, then investigate it.



```
{
    "ModelFile": "model_T1.FEM",
    "ResultsInterfaceFile": "result_R1.SIN",
    "UlsCommandFile": "SesamCore_yield.jnl",
}
```

  - In this input file, FEM model file, result file, and Sesam Core journal file are specified.
  - Here the capacity #.json file from GeniE is not necessary.

- Open **SesamCore_buckling.jnl** file, then investigate it.

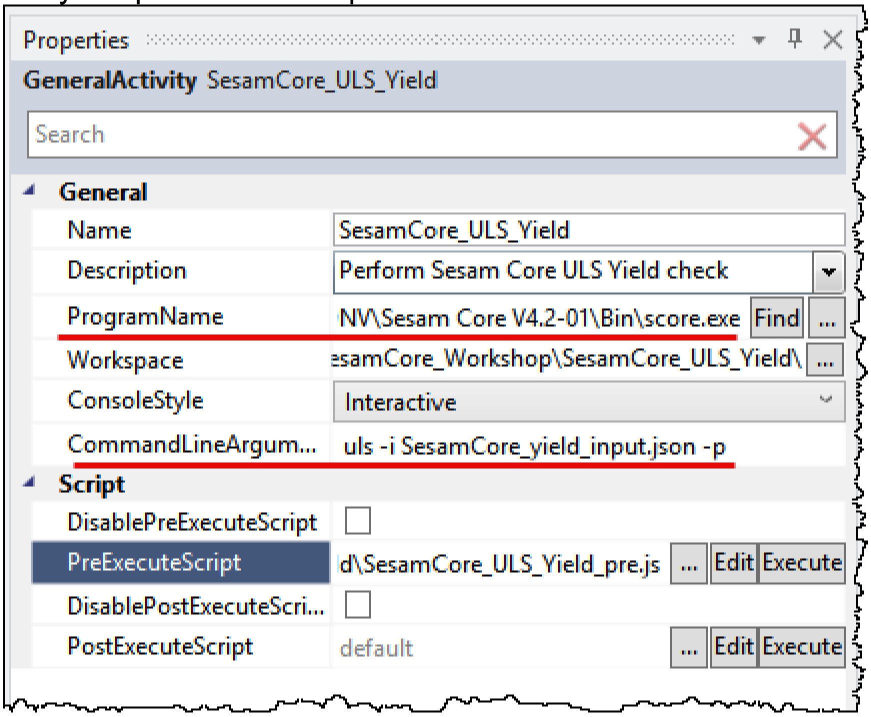


```
%
SELECT SET ELEMENTS DEFAULT INCLUDE SET AA_UpperDeck_CH4
ASSIGN MATERIAL-FACTOR 1.15
ASSIGN YIELD-STRESS-FACTOR AA_UpperDeck_CH4 1.0
RUN YIELD-CHECK yield1 'Yield Check UpperDeck port-side'
%Not yet supported:
%
%ASSIGN MATERIAL-FACTOR 1.15
```

  - `SELECT SET ELEMENTS DEFAULT INCLUDE SET AA_UpperDeck_CH4` : Select element by set
  - `ASSIGN MATERIAL-FACTOR 1.15` : Assign material factor for current selection
  - `ASSIGN YIELD-STRESS-FACTOR AA_UpperDeck_CH4 1.0`: Assign Yield Stress factor
  - `RUN YIELD-CHECK` : run yield check.
  - For more details, please refer Sesam Core User manual.

5.3 Execute Sesam Core

- To run Sesam Core, path of the Sesam Core and command line arguments should be defined.
  They're specified in "Properties" as below.



Properties

GeneralActivity SesamCore_ULS_Yield

Search

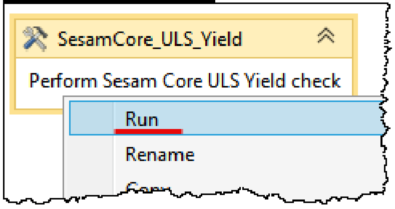
General

Name	SesamCore_ULS_Yield
Description	Perform Sesam Core ULS Yield check
ProgramName	NV\Sesam Core V4.2-01\Bin\score.exe
Workspace	esamCore_Workshop\SesamCore_ULS_Yield\
ConsoleStyle	Interactive
CommandLineArgum...	uls -i SesamCore_yield_input.json -p

Script

DisablePreExecuteScript	□
PreExecuteScript	d\SesamCore_ULS_Yield_pre.js
DisablePostExecuteScri...	□
PostExecuteScript	default

- Then program will be executed as below.
  - Command: "C:\Program Files\DNV\Sesam Core V4.2-01\Bin\score.exe" uls -i SesamCore_yield_input.json -p
  - Here "ULS" argument is to perform ULS buckling check.
  - "-i SesamCore_yield_input.json" specifying the input file of Sesam Core
  - "-p" is to perform Sesam Core Post-processing only. Without that command, Sestra will be executed simultaneously.
- Correct path of the "score.exe", if want to use different version of Sesam Core
- **Run this step** to execute Seasm Core.



SesamCore_ULS_Yield

Perform Sesam Core ULS Yield check

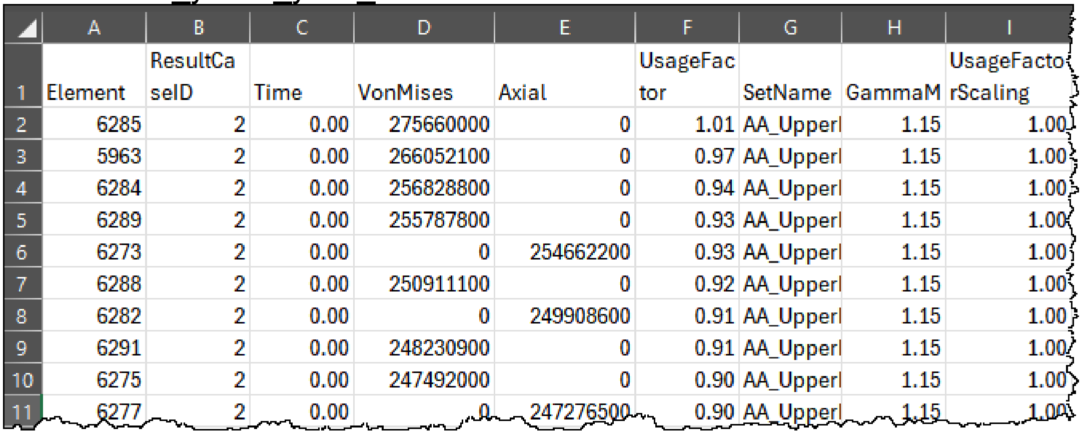
❖ If user want to run in command line (CLI), then use run_SesamCore_ULS_Yield.bat batch file in the activity folder.

5.4 Inspect the results #.lis files

After finishing Sesam Core ULS, user may check result and feed-back files

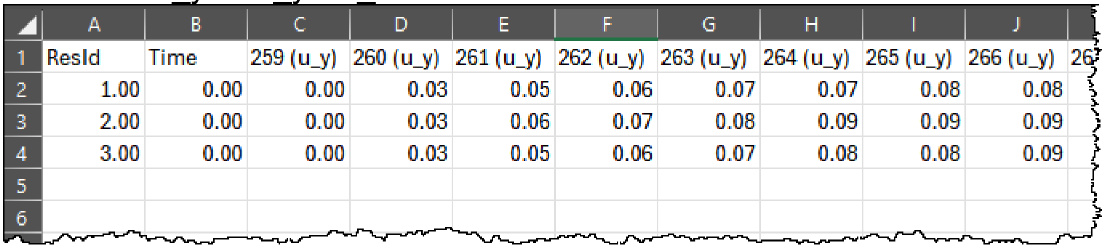
- **SCORE.MLG** : feed-back from the Sesam Core
- **#.csv files** : yield check results for each element

SesamCore_yield1_yield_max.csv



	A	B	C	D	E	F	G	H	I
1	Element	ResultCaseID	Time	VonMises	Axial	UsageFactor	SetName	GammaM	UsageFactorScaling
2	6285	2	0.00	275660000	0	1.01	AA_Upperl	1.15	1.00
3	5963	2	0.00	266052100	0	0.97	AA_Upperl	1.15	1.00
4	6284	2	0.00	256828800	0	0.94	AA_Upperl	1.15	1.00
5	6289	2	0.00	255787800	0	0.93	AA_Upperl	1.15	1.00
6	6273	2	0.00	0	254662200	0.93	AA_Upperl	1.15	1.00
7	6288	2	0.00	250911100	0	0.92	AA_Upperl	1.15	1.00
8	6282	2	0.00	0	249908600	0.91	AA_Upperl	1.15	1.00
9	6291	2	0.00	248230900	0	0.91	AA_Upperl	1.15	1.00
10	6275	2	0.00	247492000	0	0.90	AA_Upperl	1.15	1.00
11	6277	2	0.00	0	247276500	0.90	AA_Upperl	1.15	1.00

SesamCore_yield1_yield_check.csv



	A	B	C	D	E	F	G	H	I	J	
1	ResId	Time	259 (u_y)	260 (u_y)	261 (u_y)	262 (u_y)	263 (u_y)	264 (u_y)	265 (u_y)	266 (u_y)	26
2	1.00	0.00	0.00	0.03	0.05	0.06	0.07	0.07	0.08	0.08	
3	2.00	0.00	0.00	0.03	0.06	0.07	0.08	0.09	0.09	0.09	
4	3.00	0.00	0.00	0.03	0.05	0.06	0.07	0.08	0.08	0.09	
5											
6											

6 REFERENCES

- DNV-RP-C201 Buckling strength of plated structures, Sep. 2023
- DNV-OS-C101 Structural design of offshore units, Jul. 2025
- DNVGL-CG-0127 Finite element analysis, Nov 2020

About DNV

We are the independent expert in risk management and quality assurance. Driven by our purpose, to safeguard life, property and the environment, we empower our customers and their stakeholders with facts and reliable insights so that critical decisions can be made with confidence. As a trusted voice for many of the world's most successful organizations, we use our knowledge to advance safety and performance, set industry benchmarks, and inspire and invent solutions to tackle global transformations.

Digital Solutions

DNV is a world-leading provider of digital solutions and software applications with focus on the energy, maritime and healthcare markets. Our solutions are used worldwide to manage risk and performance for wind turbines, electric grids, pipelines, processing plants, offshore structures, ships, and more. Supported by our domain knowledge and Veracity assurance platform, we enable companies to digitize and manage business critical activities in a sustainable, cost-efficient, safe and secure way.