In [105…

```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import os
```

In [106…

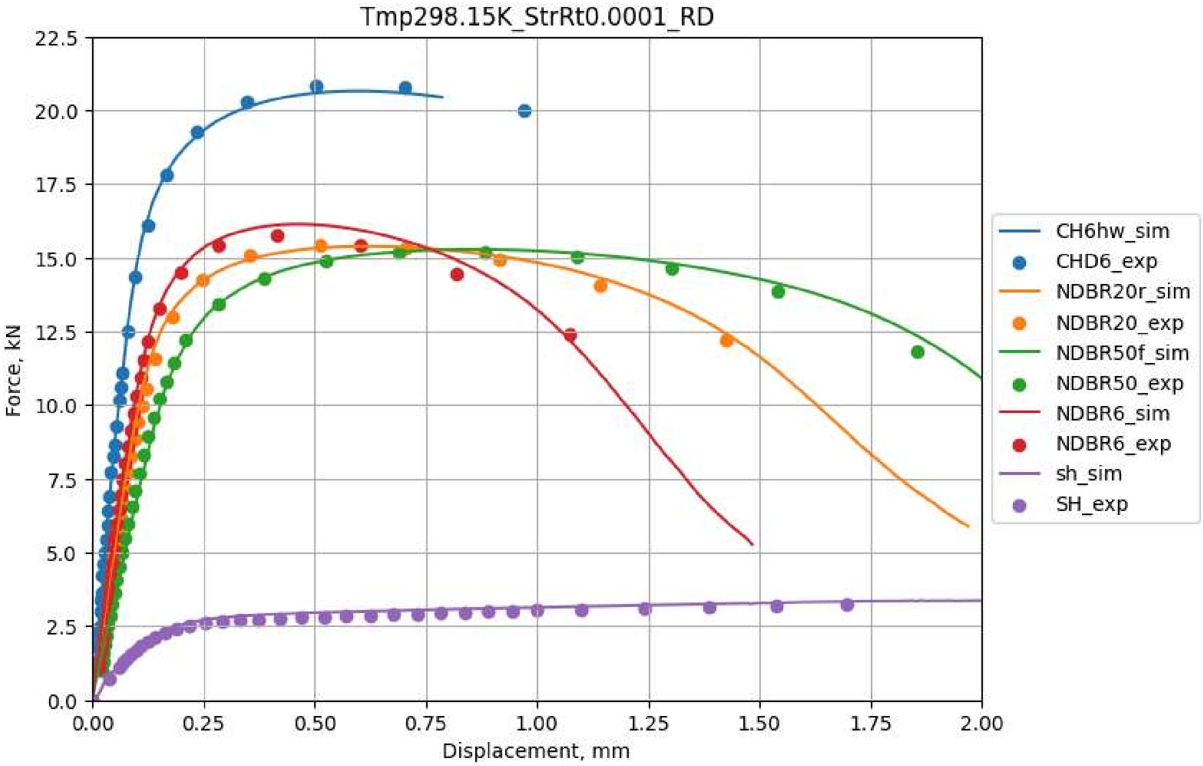
```python
def list_files(directory, extension):
    return [os.path.join(directory, f) for f in os.listdir(directory) if f.endswith

def compare_all_curve(rpt_file_paths, excel_file_paths, plot_title):
    plt.figure(figsize=(8, 6))

    for rpt_file_path, excel_file_path in zip(rpt_file_paths, excel_file_paths):
        # Reading the .rpt file with latin-1 encoding
        with open(rpt_file_path, 'r', encoding='latin-1') as file:
            rpt_content = file.readlines()

        # Parse the numeric data from the .rpt file
        rpt_data = []
        for line in rpt_content:
            if line.strip() and not line.startswith(('X', 'Displacement', 'Force'))
                values = line.split()
                if len(values) == 3:
                    try:
                        displacement = float(values[1])
                        force = float(values[2])
                        rpt_data.append([displacement, force])
                    except ValueError:
                        continue

        rpt_data = np.array(rpt_data)

        # Check if rpt_data is 2D
        if rpt_data.ndim != 2 or rpt_data.shape[1] != 2:
            raise ValueError(f"Parsed .rpt data from {rpt_file_path} is not in the

        # Convert the .rpt file force values from N to kN
        rpt_data_converted = rpt_data.copy()
        rpt_data_converted[:, 1] = rpt_data[:, 1] / 1000  # Convert N to kN

        # Load the Excel data
        excel_df = pd.read_excel(excel_file_path)

        # Extract relevant columns from the Excel file
        displacement_col_excel = excel_df.columns[0]  # First column for displaceme
        force_col_excel = excel_df.columns[1]         # Second column for force
        upper_col_excel = excel_df.columns[2]          # Third column for upper bou
        lower_col_excel = excel_df.columns[3]          # Fourth column for lower bo

        displacement_data_excel = excel_df[displacement_col_excel]
        force_data_excel = excel_df[force_col_excel]
        upper_data_excel = excel_df[upper_col_excel]
        lower_data_excel = excel_df[lower_col_excel]

        # Plot the .rpt data (converted to kN)
        plt.plot(rpt_data_converted[:, 0], rpt_data_converted[:, 1], label=f"{os.pa

        # Plot the Excel data (already in kN)
        plt.scatter(displacement_data_excel[::30], force_data_excel[::30], label=f"

        # Plot the upper and lower bounds
        #  plt.errorbar(displacement_data_excel[::30], force_data_excel[::30], yerr

    # Labels and title
    plt.title(plot_title)
    plt.xlabel("Displacement, mm")
    plt.ylabel("Force, kN")
    plt.grid(True)
    plt.legend(loc='center left', bbox_to_anchor=(1, 0.5))
    plt.xlim(left=0, right=2)
    plt.ylim(bottom=0, top=22.5)

    # Show the combined plot
    plt.show()
```

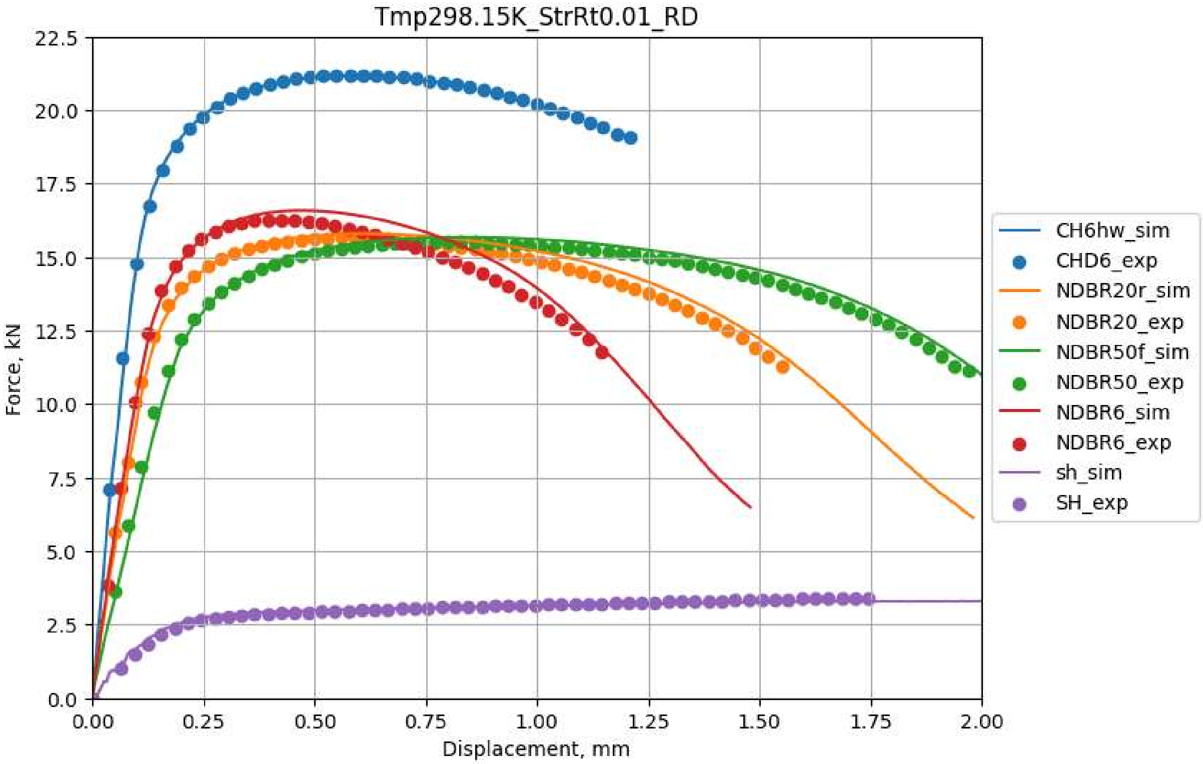
In [107…

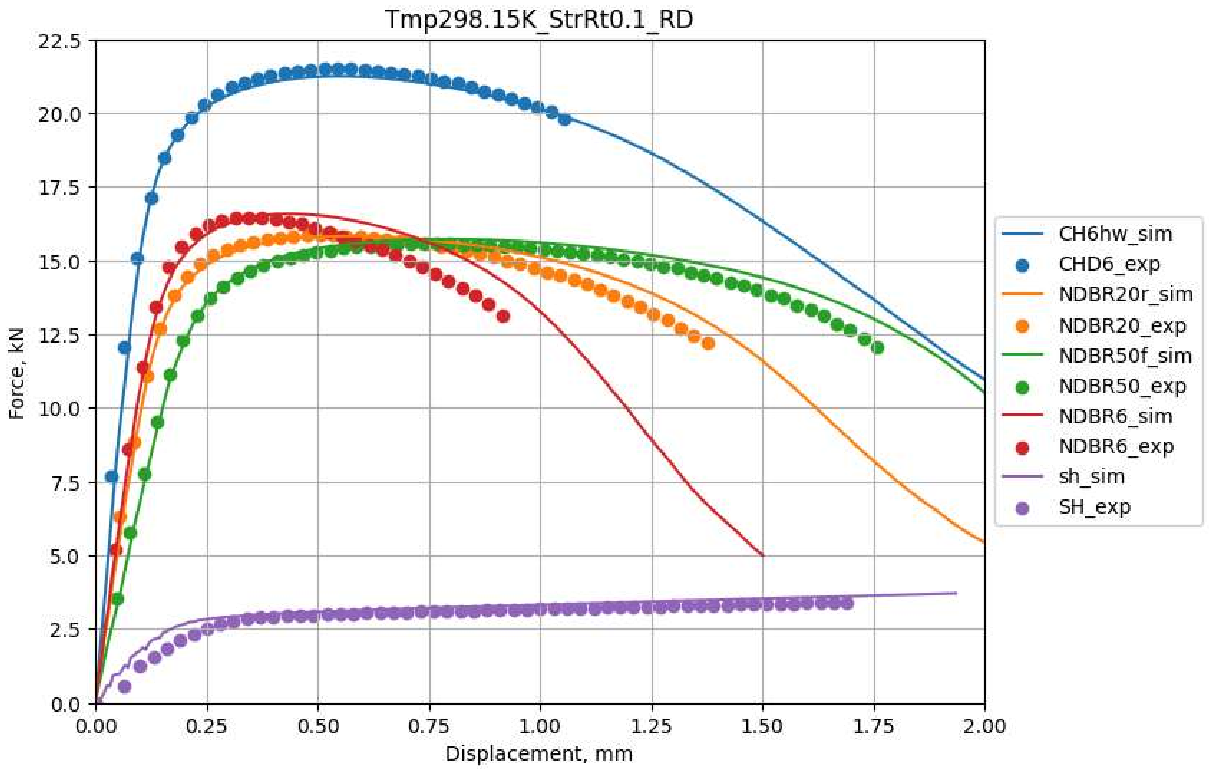
```
def process_directory(directory):
    # List all .txt and .xlsx files in the directory
    rpt_files = list_files(directory, ".txt")
    excel_files = list_files(directory, ".xlsx")

    # Ensure the lists are sorted to match pairs correctly
    rpt_files.sort()
    excel_files.sort()

    return rpt_files, excel_files
```

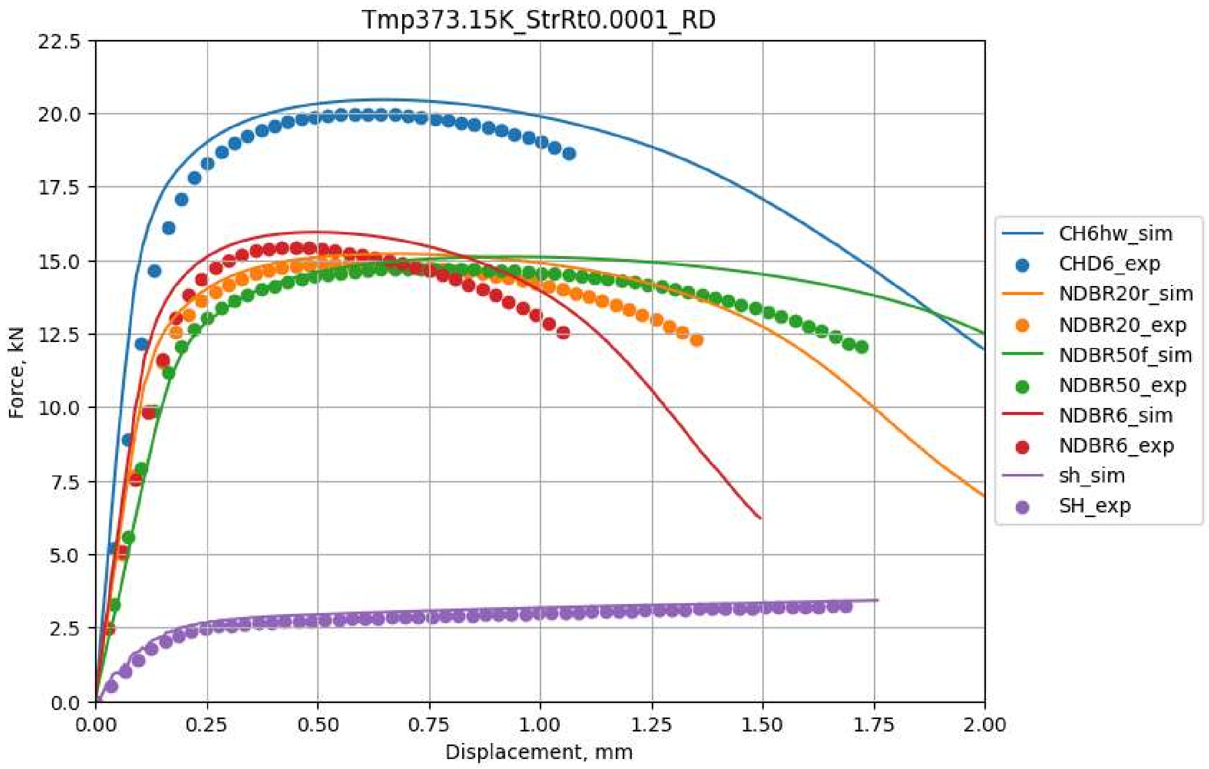
In [108…

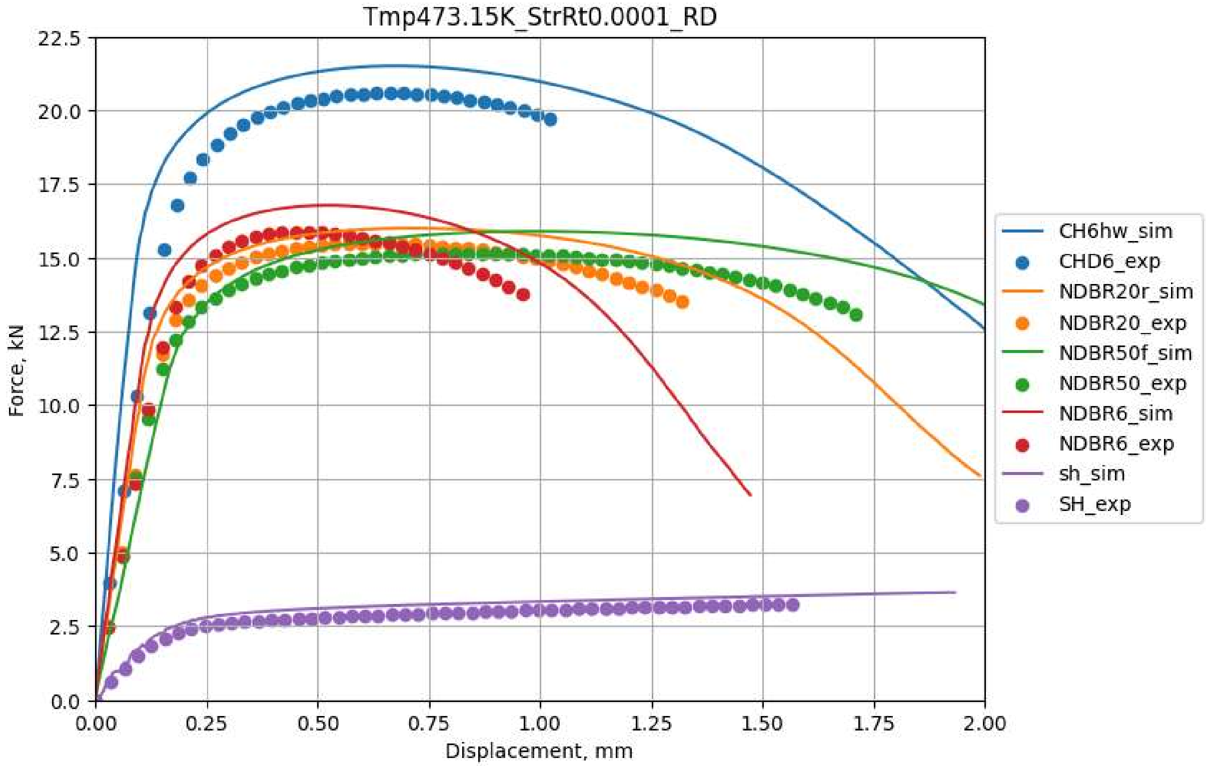
```
def process_all_subdirectories(base_directory):
    subdirectories = [os.path.join(base_directory, sub_dir) for sub_dir in os.listd

    for subdirectory in subdirectories:
        rpt_files, excel_files = process_directory(subdirectory)
        if rpt_files and excel_files:
            plot_title = os.path.basename(subdirectory)
            compare_all_curve(rpt_files, excel_files, plot_title)

# Example usage
base_directory = "C:/Users/meian/Desktop/Comp Eng Project/coe-final-project/coe-fin
process_all_subdirectories(base_directory)
```

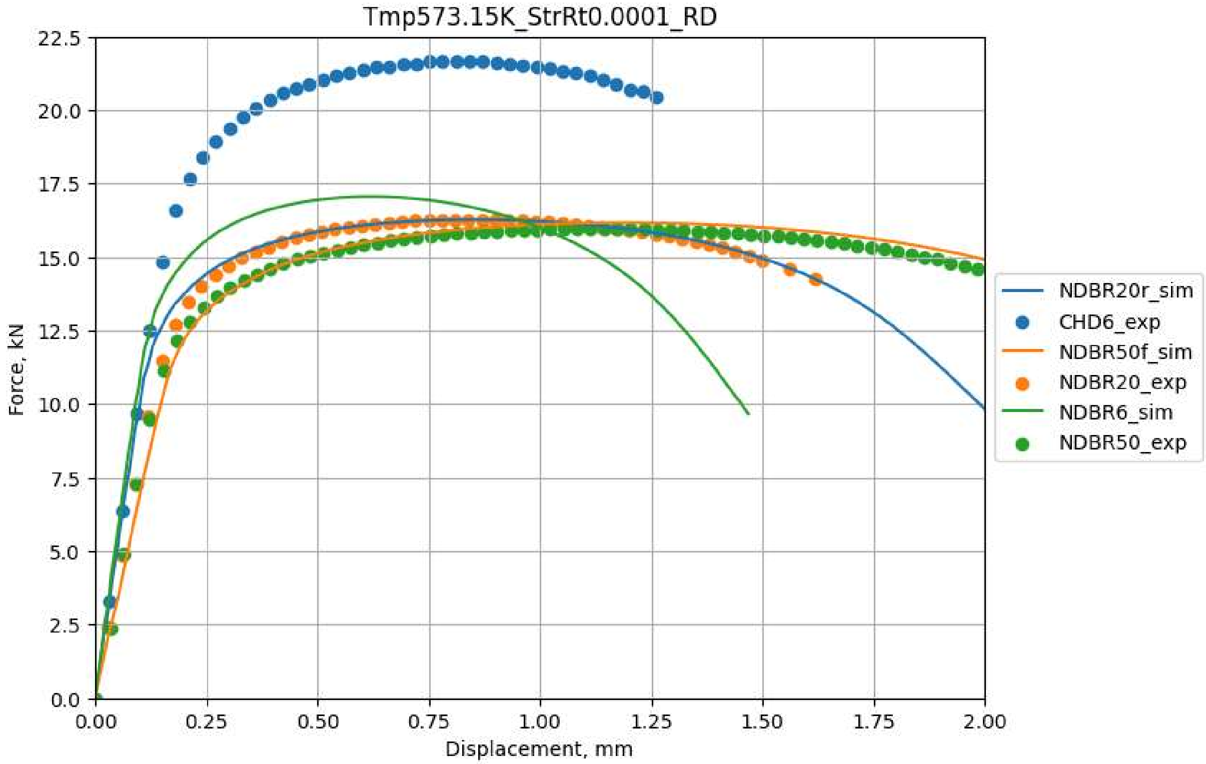

Tmp298.15K_StrRt0.0001_RD
Force, kN
Displacement, mm
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
22.5
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
CH6hw_sim
CHD6_exp
NDBR20r_sim
NDBR20_exp
NDBR50f_sim
NDBR50_exp
NDBR6_sim
NDBR6_exp
sh_sim
SH_exp

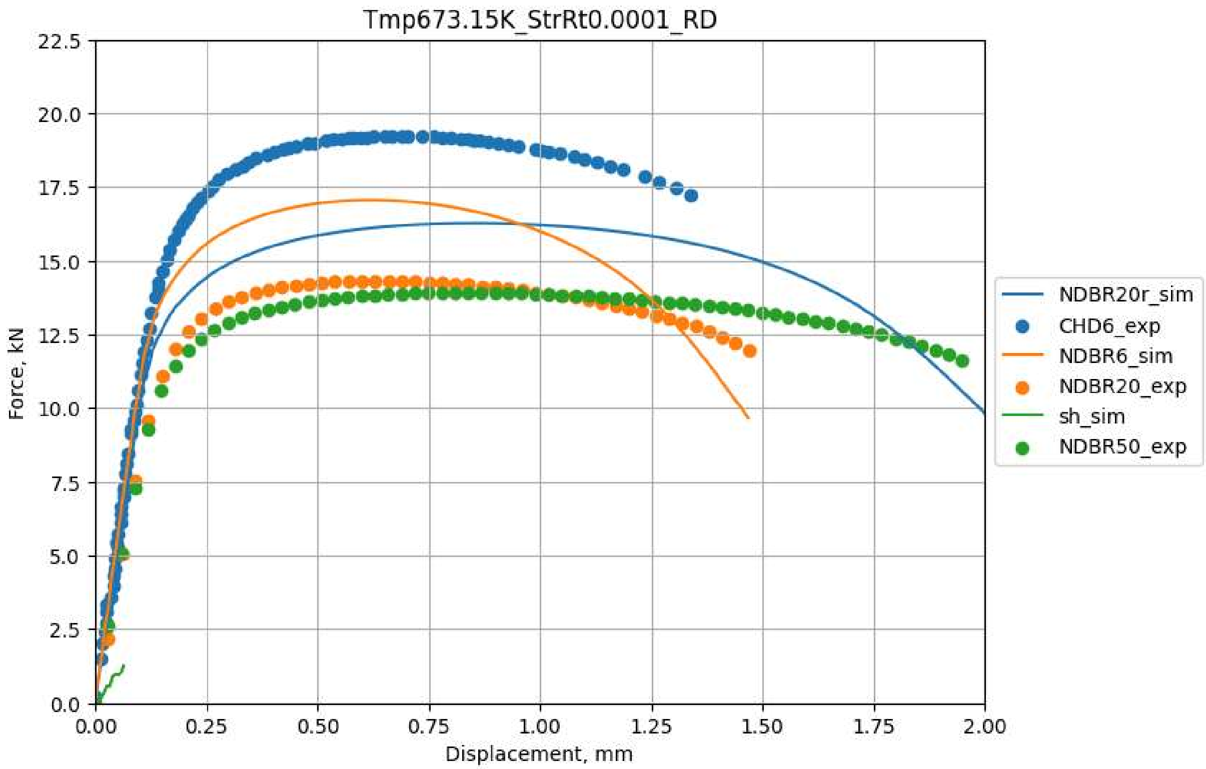

Tmp298.15K_StrRt0.01_RD
Force, kN
Displacement, mm
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
22.5
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
CH6hw_sim
CHD6_exp
NDBR20r_sim
NDBR20_exp
NDBR50f_sim
NDBR50_exp
NDBR6_sim
NDBR6_exp
sh_sim
SH_exp


Tmp298.15K_StrRt0.1_RD
Force, kN
Displacement, mm
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
22.5
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
CH6hw_sim
CHD6_exp
NDBR20r_sim
NDBR20_exp
NDBR50f_sim
NDBR50_exp
NDBR6_sim
NDBR6_exp
sh_sim
SH_exp


Tmp373.15K_StrRt0.0001_RD
Force, kN
Displacement, mm
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
22.5
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
CH6hw_sim
CHD6_exp
NDBR20r_sim
NDBR20_exp
NDBR50f_sim
NDBR50_exp
NDBR6_sim
NDBR6_exp
sh_sim
SH_exp


Tmp473.15K_StrRt0.0001_RD
Force, kN
Displacement, mm
22.5
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
CH6hw_sim
CHD6_exp
NDBR20r_sim
NDBR20_exp
NDBR50f_sim
NDBR50_exp
NDBR6_sim
NDBR6_exp
sh_sim
SH_exp


Tmp573.15K_StrRt0.0001_RD
Force, kN
Displacement, mm
22.5
20.0
17.5
15.0
12.5
10.0
7.5
5.0
2.5
0.0
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
NDBR20r_sim
CHD6_exp
NDBR50f_sim
NDBR20_exp
NDBR6_sim
NDBR50_exp


Tmp673.15K_StrRt0.0001_RD
Force, kN
Displacement, mm
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
22.5
0.00
0.25
0.50
0.75
1.00
1.25
1.50
1.75
2.00
NDBR20r_sim
CHD6_exp
NDBR6_sim
NDBR20_exp
sh_sim
NDBR50_exp