The following steps need to be followed to run the files:

1. **HVDC-RSCAD Model:**

**Step 1:** Download the files from the git directory.

**Step 2**: Extract the zip “HVDC-RTDS-Models.zip” in a temporary folder.

**Step 3:** From the extracted folder, copy the “CSE\_MODEL\_FX2.0” folder, and paste it to your RSCAD user directory (example: “C:\RSCAD\RTDS\_USER\_FX\fileman\”).

**Step 4:** Now open the RSCAD FX2.0, and load the desired models from the CSE\_MODEL\_FX2.0 folder from your directory. I am considering the “CIGRE\_4TERMINAL\_525KV\_RTS\_MODEL\_3D\_check.rtfx” test case for this explanation.

**Step 5:** Upon loading the above file, you will see the following Draft tab.

Modify the model to incorporate the neural network library(see section B).

**Step 6:** Compile the file on a rack with suitable cores, in this case, a minimum of seven cores.

**Step 7**: Now go to the Runtime tab.

**Step 8:** Now open and run the script “Parameter\_settting”. This will configure the draft variables in the runtime.

**Step 9:** Now open and run the script “starting\_squence\_5T\_file”. This will start the simulation with a starting sequence mentioned in the paper. Once the script is completed now, you can begin your study.

1. **Modification in Model for NN Application**

Add a switch to include NN output for both active and reactive power

A diagram of a circuit

Description automatically generatedA diagram of a circuit

Description automatically generated

1. **C-Buider**
2. Put all the c-files [.c, .h,] except the def files in Bin-CMODELSOURCE (.h file simply contains definitions of inputs and outputs, e .c file should include the functionality of the component, see details in Chapter 6 of thesis) [C:\Users\rashmiprasad\Documents\RSCAD\BIN\CMODEL\_SOURCE]
3. Put the def files on -Component folder C:\Users\rashmiprasad\Documents\RSCAD\ULIB\COMPONENTS
4. In the toolbar: Go to Launch- Component Builder- File Open- .def file from

C:\Users\rashmiprasad\Documents\RSCAD\ULIB\COMPONENTS

1. A screenshot of a computer

   Description automatically generatedOpen C File Associations and select and compile the codes
2. Goto-Draft-add-component-User lib-add that .def as a component in the draft.

**Test Case1:**Offline Training-2

1. Saving files: Get the Data from the runtime.- plot the inner current control and outer voltage control PI input and output parameters- [(Perr\_CSA4P, PIpwr\_CSA4P), (qerr\_CSA4P, IQREFq\_CSA4P), (iderr\_CSA4P, outpi1\_t\_CSA4P), (iqerr\_CSA4P, outpi2\_t\_CSA4P] Use the script file to run multiple case to save data by varying P and Q
2. Code
3. Generation of weight and updating

**Test Case2:**Online Training

NNUpdate1: layer6

NNUpdate2: 10 layers possible

Less layers work for PI modelling, even two layers is able to provide a good estimate.

**Script to collect data from RSCAD:**

int b=1;

function savefile2(int b)

{

fprintf(stdmsg,"Saving plot data for positive idmode Case Number %d\n",b);

return();SavePlotToCSV "collection\_data","C:\Users\rashmiprasad\OneDrive - Delft University of Technology\Desktop\Bara\HVDC-RTDS-models-main\CSE\_MODEL\_FX2.0\Final\_Data\ErrData\new\Filename\_"::itoa(b)::".csv";

}

//Example Function

float var = rand(1)\*1000;

fprintf(stdmsg,"the random number %f\n" , var);

float start\_value = 100.0; // Starting value for Pref\_CSA5P

float end\_value = 1000.0; // Ending value for Pref\_CSA5P (example value, adjust as needed)

float increment = 1.0; // Increment step for Pref\_CSA5P

int num\_steps = (end\_value - start\_value) / increment;

float start\_valueq = 100.0; // Starting value for Pref\_CSA5P

float end\_valueq = 1000.0; // Ending value for Pref\_CSA5P (example value, adjust as needed)

float incrementq = 1.0;

int num\_stepsq = (end\_valueq - start\_valueq) / incrementq;

int i=0;

// Use a for loop to change the Pref\_CSA5P variable and save the plot

//for (int i = 0; i <= num\_steps; i++)

{

// Set the value of Pref\_CSA5P

float current\_value = start\_value + (i \* increment);

float current\_valueq = start\_valueq + (i \* incrementq);

SetSlider "Subsystem #1 : CTLs : Inputs : Pref\_CSA4P" = current\_value;

SUSPEND 1.0;

UpdatePlots;

savefile2(2\*i + 1);

SetSlider "Subsystem #1 : CTLs : Inputs : Qref\_CSA4P" = current\_valueq;

SUSPEND 1.0;

UpdatePlots;

savefile2(2\*i + 2);

SUSPEND 0.3;

}

//Stop;