**Vérification des résultats d’autocodage**

Un fois le projet créé et modifié puis lancé, vous devriez avoir un dossier « OUTPUTS\_CODE » contenant tous les fichiers de sortie issus du programme C++. Nous créons ensuite un script appelé « Test\_autocodage\_mon\_bloc.m », ce fichier se trouve dans chaque dossier relatif au bloc autocodés pour raison de commodité. Le script est architecturé de la manière suivante :

* Initialisation des paramètres de simulation
* Création de la structure de sorties en simulation, contenant tous les fichiers textes chargés se trouvant dans le dossier « OUTPUTS »
* Création de la structure de sorties en autocodage, contenant tous les fichiers textes chargés se trouvant dans le dossier « OUTPUTS\_CODE »
* Tracé de chaque sortie pour comparer la réponse Simulink et la réponse Autocodée
* Tracé de chaque erreur entre la sortie autocodée et la sortie Simulink. L’erreur est calculée comme erreur absolue en pourcent par rapport à la valeur max absolue de la sortie Simulink

Voici un exemple de script utilisé pour un bloc :

% - - - - - TEST AUTOCODAGE CLUTCH - - - - - %

% - - - - - IL FAUT AVOIR CREE LES FICHIERS OUTPUTS\_code - - - - - %

% - - - - - Ici ou est en 200s de simu avec un pas de 0.01 - - - - - %

close all

clear all

tfin = 200;

pas = 0.01;

t = 0:pas:tfin;

% Sortie de base %

% On cree une structure dans laquelle on charge les fichiers de sortie de

% la simulation

OB\_CLU = struct;

OB\_CLU.FA\_N\_ICE = load('OUTPUTS\FA\_N\_ICE.txt');

OB\_CLU.FA\_TQ\_CLU = load('OUTPUTS\FA\_TQ\_CLU.txt');

OB\_CLU.FA\_CLU\_STUCK = load('OUTPUTS\FA\_CLU\_STUCK.txt');

% Sortie codee %

% On cree une structure dans laquelle on charge les fichiers de sortie

% issus de l'autocodage

OB\_CLU\_code = struct;

OB\_CLU\_code.FA\_N\_ICE\_code = load('OUTPUTS\_CODE\FA\_N\_ICE\_code.txt');

OB\_CLU\_code.FA\_TQ\_CLU\_code = load('OUTPUTS\_CODE\FA\_TQ\_CLU\_code.txt');

OB\_CLU\_code.FA\_CLU\_STUCK\_code = load('OUTPUTS\_CODE\FA\_CLU\_STUCK\_code.txt');

% Affichage - Comparaison %

figure

plot(t,OB\_CLU.FA\_N\_ICE)

hold on

plot(t,OB\_CLU\_code.FA\_N\_ICE\_code)

title('FA\_N\_ICE')

legend('simulation','autocodage')

figure

plot(t,OB\_CLU.FA\_TQ\_CLU)

hold on

plot(t,OB\_CLU\_code.FA\_TQ\_CLU\_code)

title('FA\_TQ\_CLU')

legend('simulation','autocodage')

figure

plot(t,OB\_CLU.FA\_CLU\_STUCK)

hold on

plot(t,OB\_CLU\_code.FA\_CLU\_STUCK\_code)

title('FA\_CLU\_STUCK')

legend('simulation','autocodage')

figure

subplot(1,3,1)

plot(t,abs(OB\_CLU.FA\_N\_ICE-OB\_CLU\_code.FA\_N\_ICE\_code)\*100/max(abs(OB\_CLU.FA\_N\_ICE)))

title('Erreur FA\\_N\\_ICE')

subplot(1,3,2)

plot(t,abs(OB\_CLU.FA\_TQ\_CLU-OB\_CLU\_code.FA\_TQ\_CLU\_code)\*100/max(abs(OB\_CLU.FA\_TQ\_CLU)))

title('Erreur FA\\_TQ\\_CLU')

subplot(1,3,3)

plot(t,abs(OB\_CLU.FA\_CLU\_STUCK-OB\_CLU\_code.FA\_CLU\_STUCK\_code)/max(abs(OB\_CLU.FA\_CLU\_STUCK)))

title('Erreur FA\\_CLU\\_STUCK')