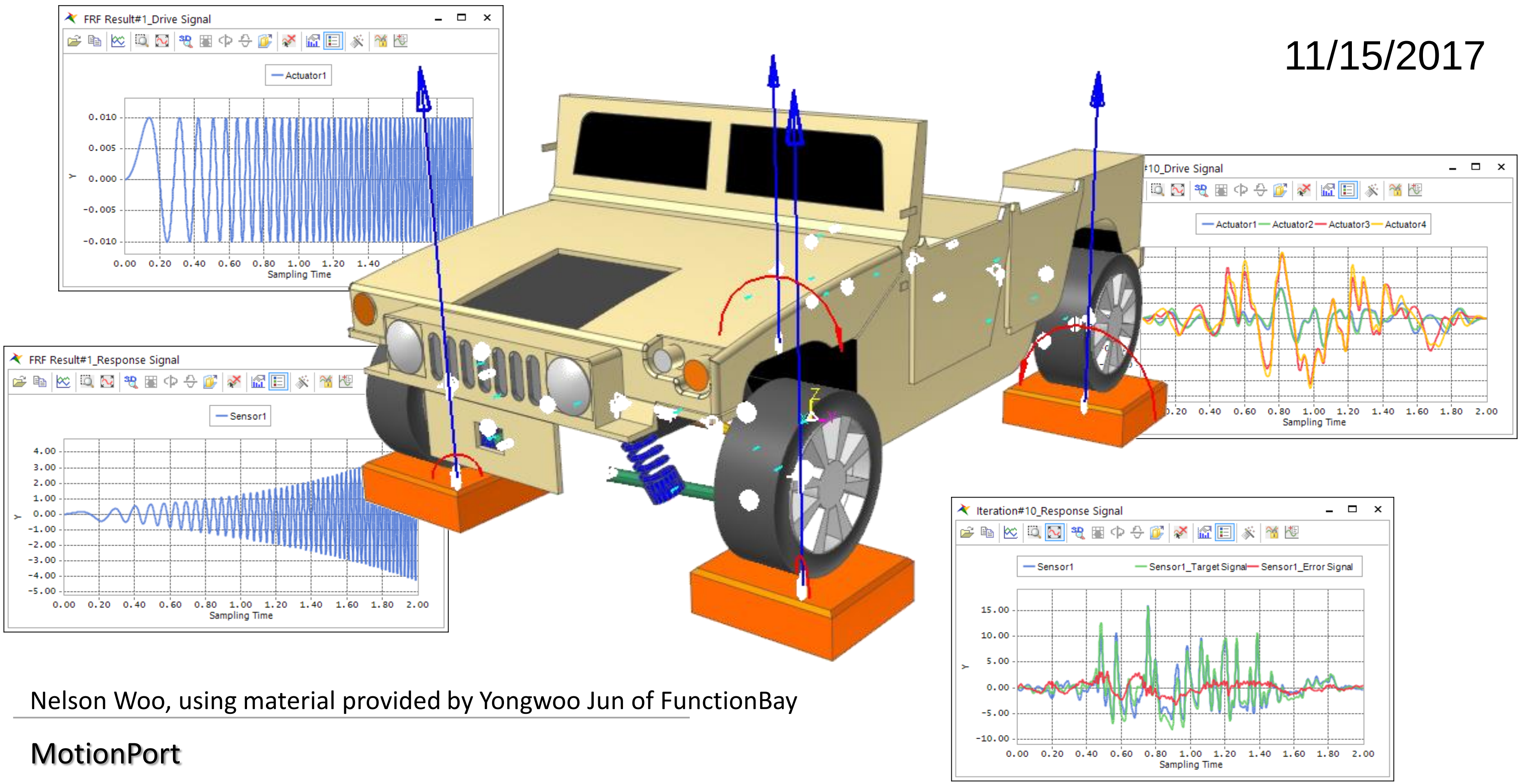
RECURDYN

RecurDyn/TSG (Time Signal Generator)

11/15/2017

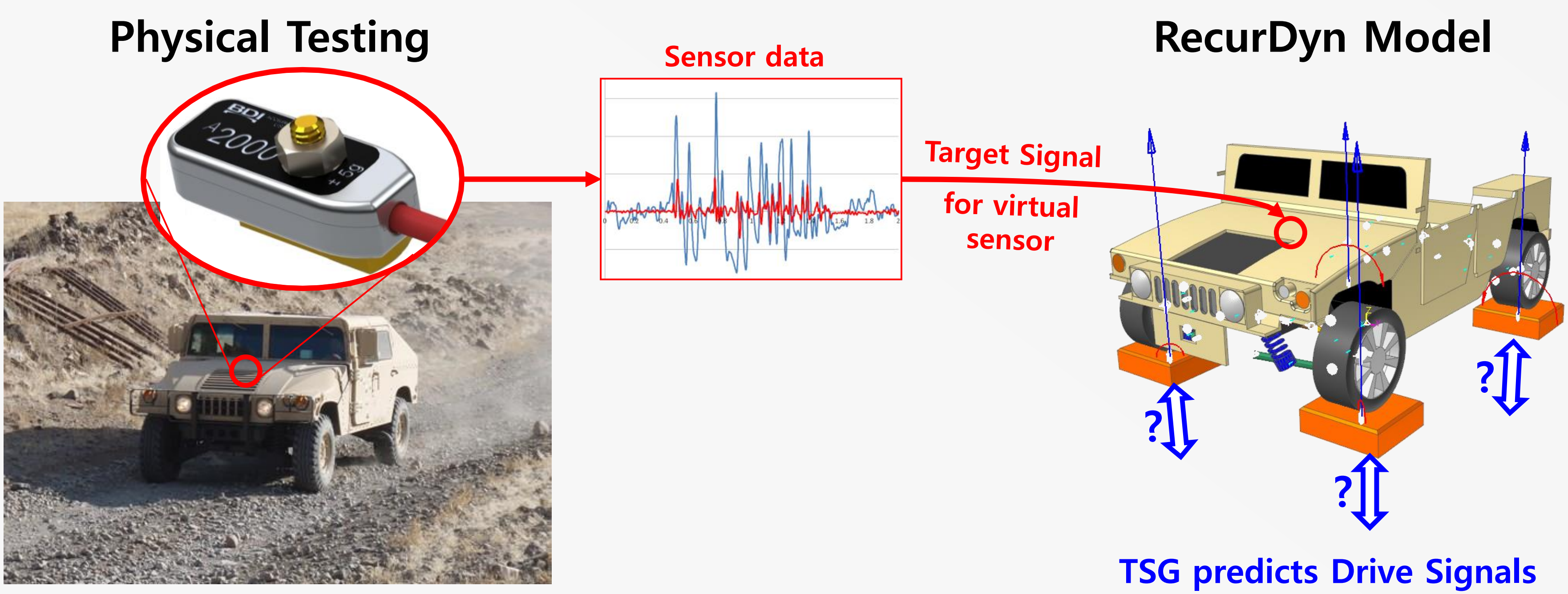


Nelson Woo, using material provided by Yongwoo Jun of FunctionBay

MotionPort

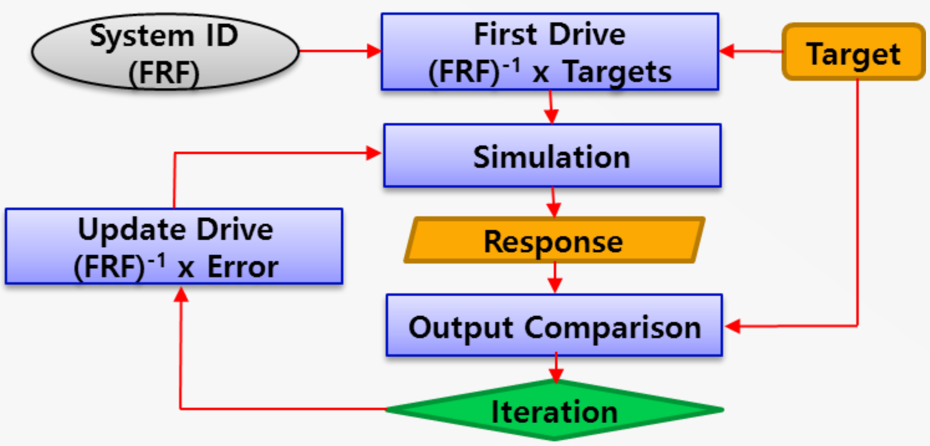
Overview

- RecurDyn/TSG (Time Signal Generator) allows analysts to use data from physical testing to replicate test conditions in an analogous CAE (Computer Aided Engineering) model.
- Data from physical testing is limited by the type and number of sensors that can be instrumented, leading to incomplete picture of what happens during the test.
- RecurDyn/TSG can take available test outputs (i.e. vehicle chassis acceleration from accelerometers) and backsolve to find the test inputs (i.e. displacement to be enforced at the tires), which can be difficult to obtain due to physical limitations of test instrumentation.
- For automotive applications, removes the requirement for virtual terrain, tire, and driver modeling, which can be difficult to characterize correctly.
- Complex nonlinear behavior of the model is automatically taken into account as RecurDyn/TSG solves for the input signals.



Procedure

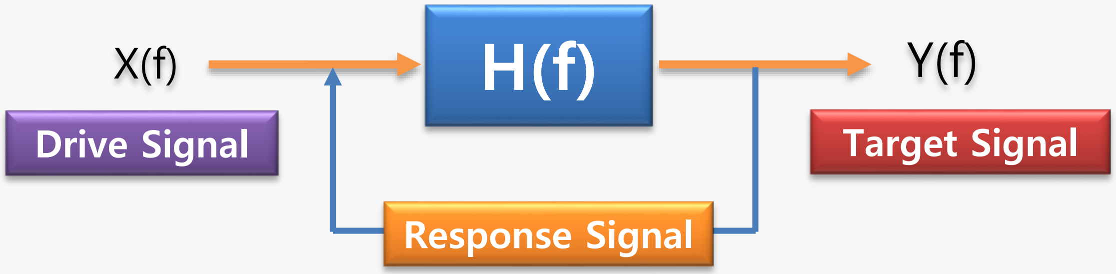
1. Create actuators to apply the displacement/load input to the model.
2. Create virtual sensors in the RecurDyn model where the actual sensors are located in the physical model.
3. Import the physical test data to become the desired output for the virtual sensors. This is the **Target Signal**.
4. RecurDyn/TSG runs a frequency response analysis to determine the system's Frequency Response Function (FRF).
5. Using the FRF and the Target Signal, RecurDyn/TSG then iteratively backsolves to find the proper **Drive Signal** for the actuators to reproduce the Target Signal from the virtual sensors.
6. If the responses of the sensors are similar to the measured Target Signals, then RecurDyn model can be regarded that it is actuated similarly to the real system.



TSG Terminology

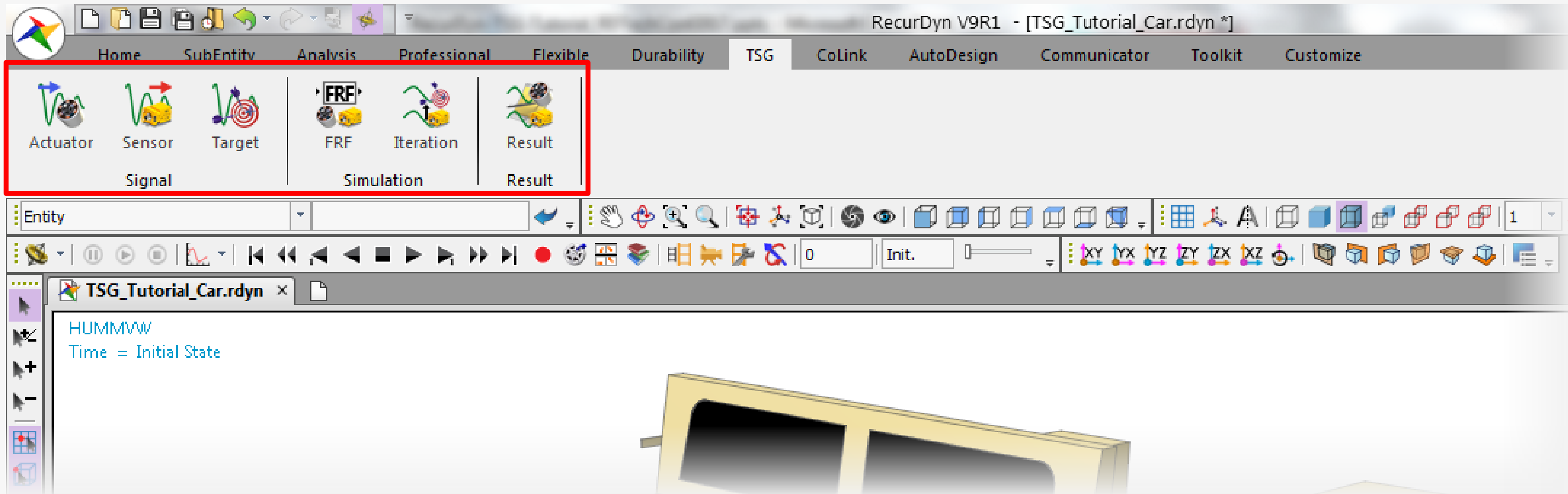
Actuator:
- Drives the system at a various location of the system.
- It is defined using a Joint Motion or a Force in an MBD model
- This is done using the function **TACT()** in an expression.

Sensor:
- The response of a system is calculated using sensors.
- Eventually, the responses of the sensors should match the user-defined Target Signals closely.



Drive Signals: **Inputs** to MBD model at the locations of the virtual **Actuators**.
Response Signals: **Outputs** of RecurDyn at the location of the virtual **Sensors.**
Target Signals: **User-Defined Signals** that were obtained from physical testing. RecurDyn varies the Drive Signals iteratively until the Response Signals match the Target Signals.

TSG Function Layout



The RecurDyn/TSG functions are arranged in the order of the user workflow:

1. Define the **Actuator**s as forces or motion assigned to joints.
2. Define the **Sensor**s as measurements of displacement, velocity, acceleration, force, or torque at a specific location.
3. Generate the **Target Signal** from imported data.
4. Obtain the **FRF** (Frequency Response Function) by running a frequency response analysis.
5. Perform an **Iteration** of simulations to solve for the Drive Signals that make the Response Signals match the Target Signal.
6. Review the **Results**, observing how closely the Response Signals match the Target Signals.

RecurDyn/TSG Tutorial

Tutorial (1)

1. Actuator

① Actuator: define the number of actuators.

a. As shown in the figure below, select the **Actuator** icon to open Actuator List dialog.

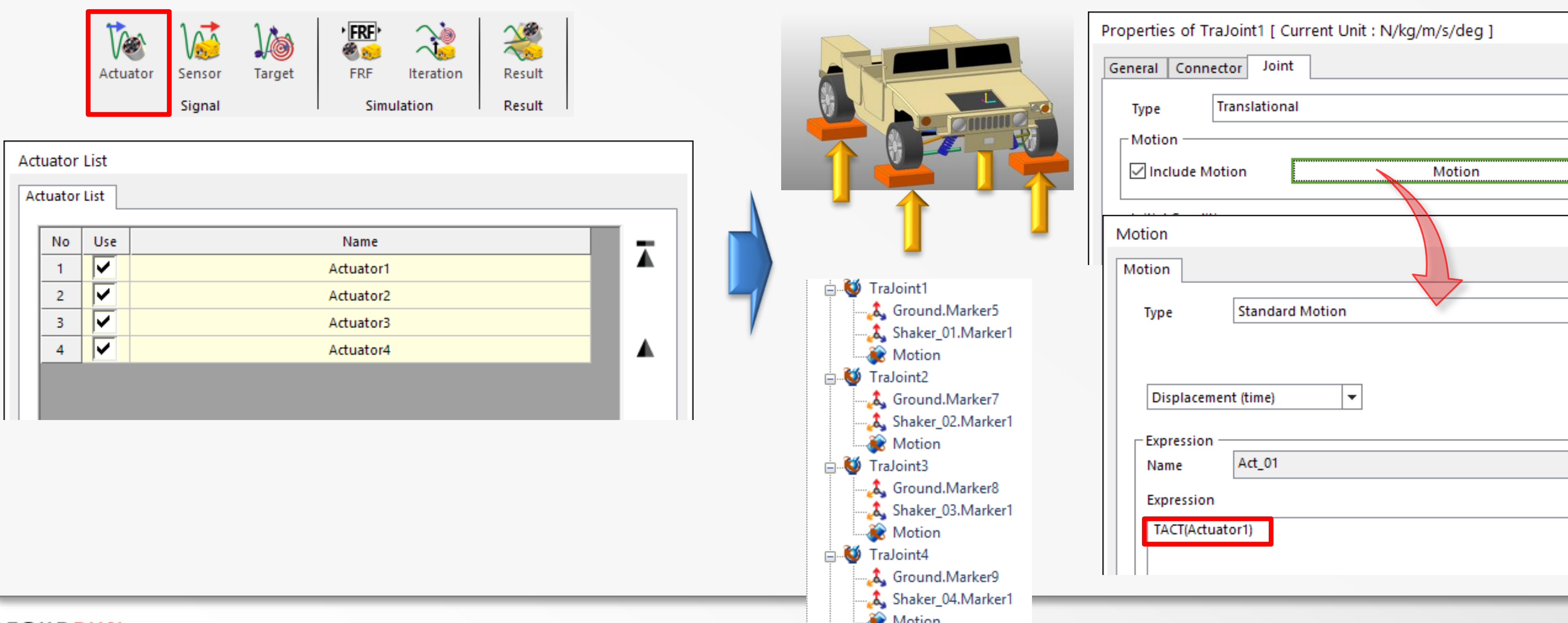
b. Create 4 actuators using **Add** button.

② Apply the actuators to the joints.

a. The actuators are assigned to Joint Motion or Force using the function expression TACT().

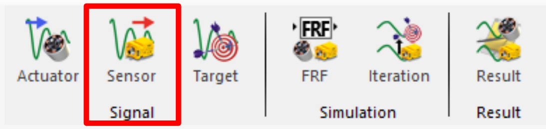
- **Apply the below functions to TraJoint1-TraJoint4 as joint motions (Displacement type).**
- **Use TACT(Actuator1), TACT(Actuator2), TACT(Actuator3), and TACT(Actuator4).**

b. In this tutorial, the actuators will move 4 shakers below each tire up and down.



Tutorial (2)

1. Sensor



① **Sensor**: The response of the simulation which will be compared with the Target Signal.

a. As shown in the below figure, select the **Sensor** icon to open Sensor List Dialog.

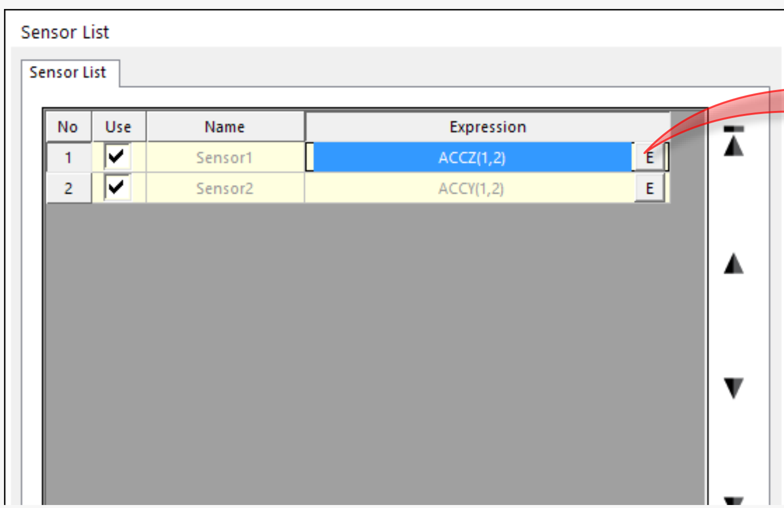
b. **Add** 2 sensors in Sensor List dialog.

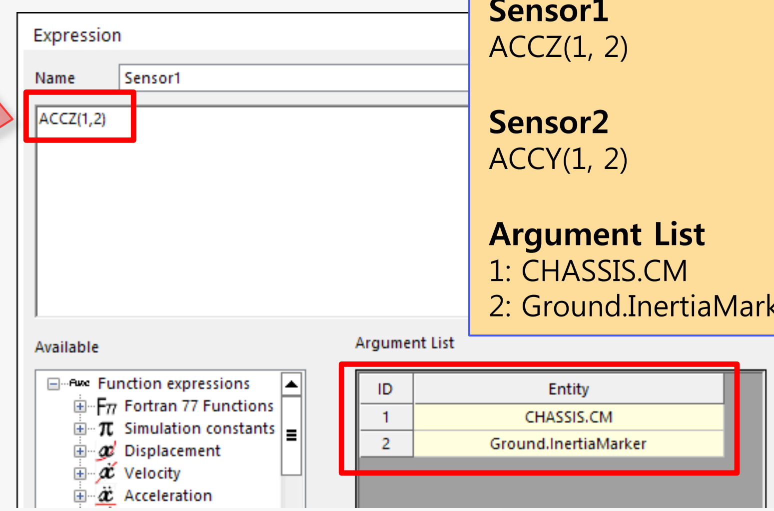
② **Define the function expressions for the sensors**

a. Any function expression can be used for sensors.

- Acceleration(ACCX, ACCY, ACCZ), Velocity(VX, VY, VZ), Disp.(DX, DY, DZ)
- Force(FX, FY, FZ, TX, TY, TZ), Stress(SX, SY, SZ), Strain(EX, EY, EZ), Etc.

b. In this tutorial, Z-Acceleration and Y-Acceleration of CM of Chassis will be used.





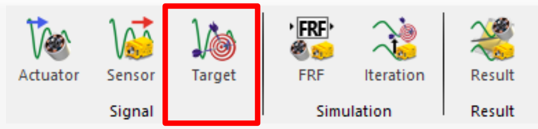
Sensor1
ACCZ(1, 2)

Sensor2
ACCY(1, 2)

Argument List
1: CHASSIS.CM
2: Ground.InertiaMarker

Tutorial (3)

1. Target (1)



① **Target**: User-defined input data.

a. Time-dependent continuous data set measured from experiment or simulation. Performance index of RecurDyn/TSG.

b. After importing measured data, Target data needs to be regenerated (into a .target file).

② **Import csv file**

a. .csv file (text file) is used.

b. The number of Target Data in .csv file is dependent on the number of Sensors.

c. The sequence of the data in. csv file must be, *time1, data1, time2, data2, …*

d. In this tutorial, there are 2 sensors, so 4 data columns must be written in .csv file as shown in the below figure.

- Even if the time data is duplicated, it should be written respectively.
- The Target Data must be written according to the sequence of the Sensors

time1	data1	time2	data2
0	0	0	0
0.001	-0.0004	0.001	0.000186
0.002	-0.00303	0.002	0.001564
0.003	-0.01027	0.003	0.009046
0.004	-0.02232	0.004	0.033633
0.005	-0.03801	0.005	0.080458
0.006	-0.05629	0.006	0.131874

Target of Sensor1 Target of Sensor2

Tutorial (4)

1. Target (2)

① Tips to generate Target Data

a. The data measured from experiment usually includes high-frequency data as well as low-frequency data.

b. The high-frequency data can cause noise and error during simulation using TSG.

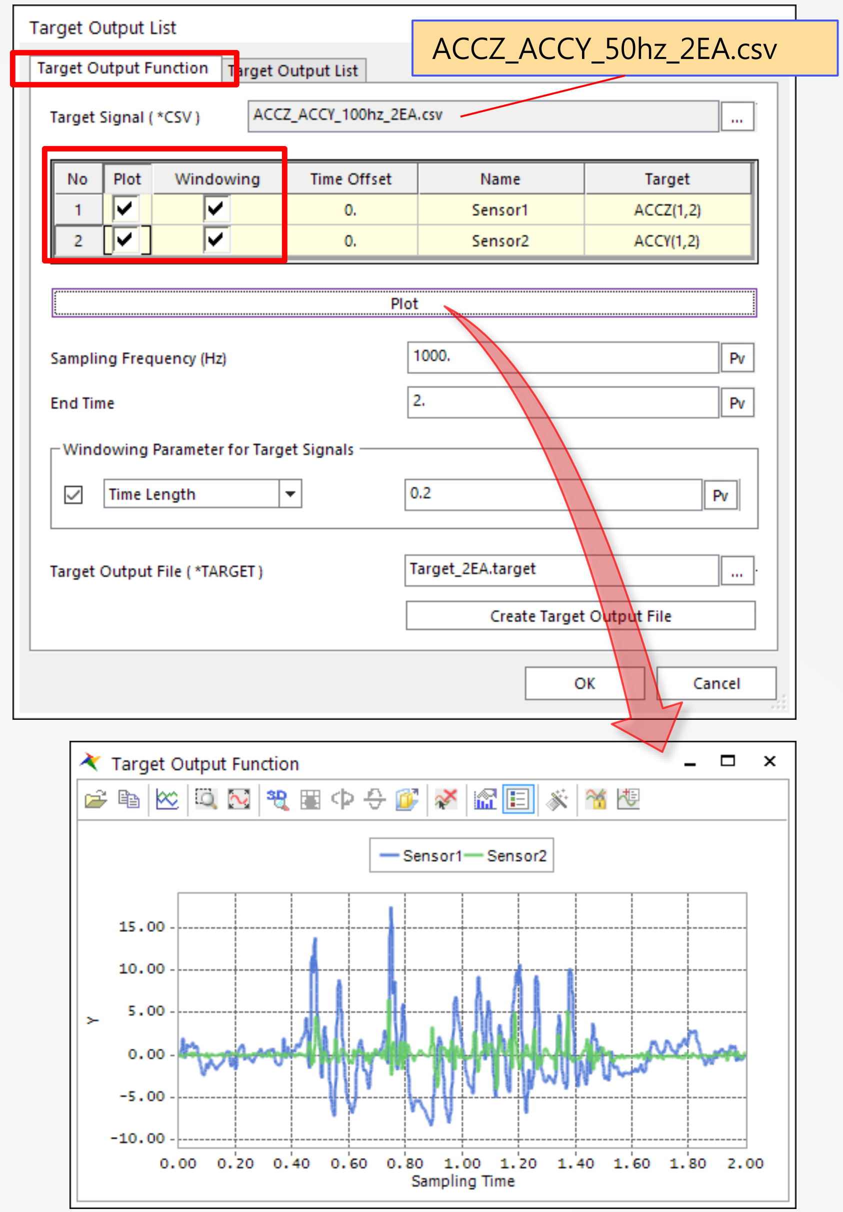
c. So it is recommended to filter the data using Low Pass Filter so that the filtered data can include the signal below 50~100Hz when you generate *.csv file.

- You can use Low Pass Filter in RecurDyn/Plot.
- The sample file of this tutorial, ACCZ_ACCY_50hz_2EA.csv includes the signal below 50Hz.

② Import .csv file

a. Import .csv file in **Target Output Function** tab of Target Output List dialog.

b. You can **plot** the Target Data for Sensor1 and Sensor2.



Tutorial (5)

1. Target (3)

① Sampling Frequency

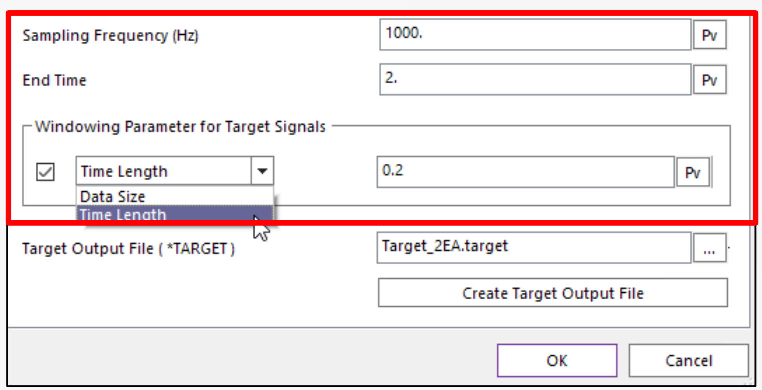
a. The number of data per 1 second. 1000 is used in this tutorial.
   - If Simulation End Time is 2 sec, the number of data must be 2000.

b. Since the number of data in .csv file doesn't match the required number, you must re-generate the data file for the given sampling frequency and end time.

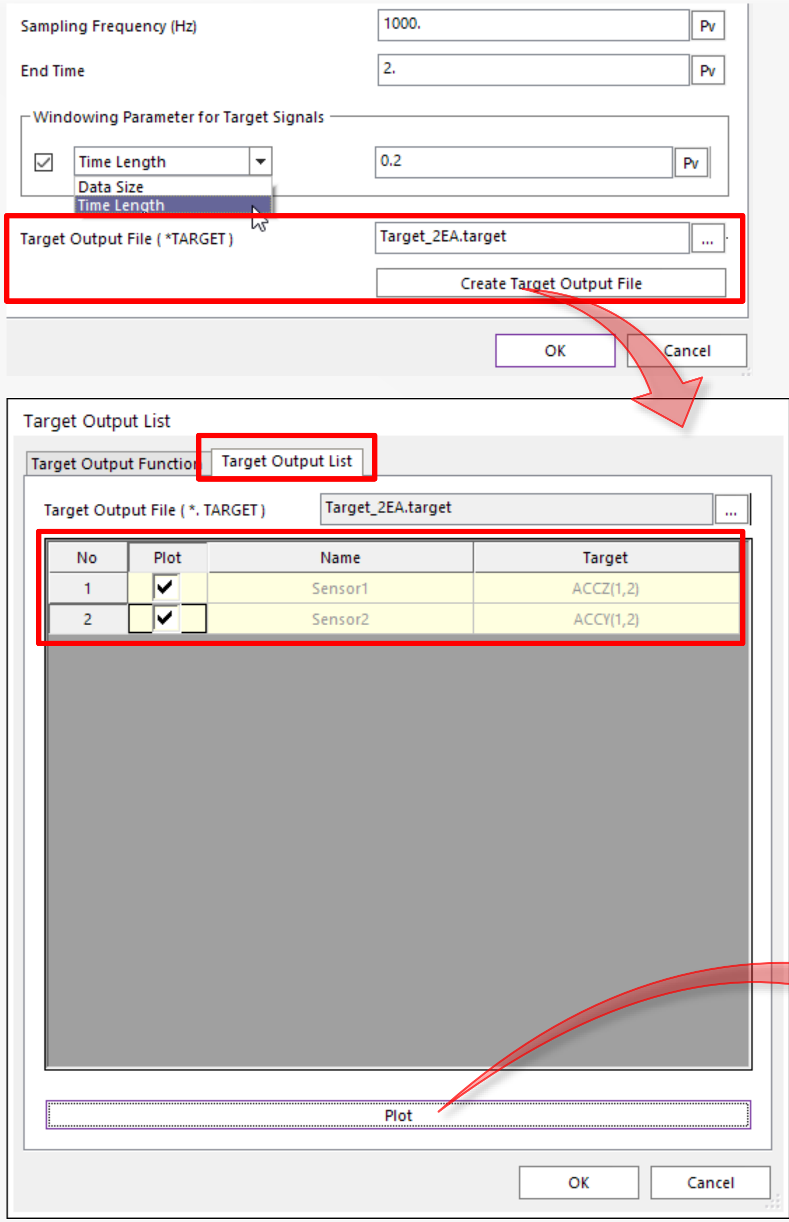
c. You will create *.target data in the next page.

② Windowing Parameter for Target Signals

a. When the Time Signal is converted to frequency signal using Fourier Transform, the initial signal and the final signal is set to zero to minimize the error.
   - Windowing is applied about 10% of the entire time.
   - In this tutorial, 0.2 with Time Length type is used.



Tutorial (6)



1. Target (4)

① Create Target Data

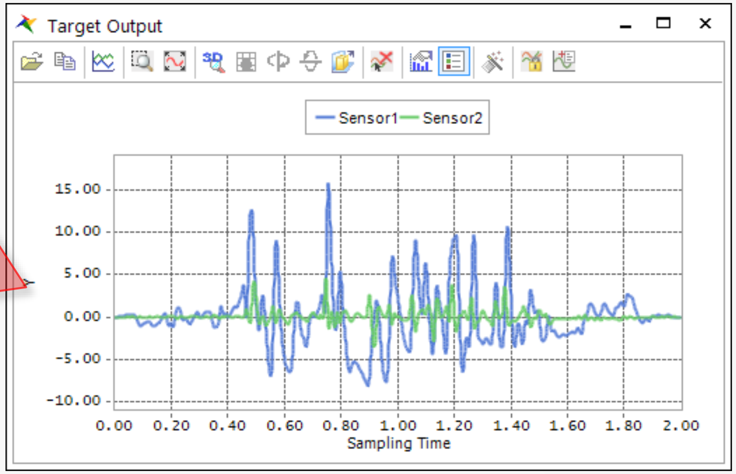
a. Create .target file from .csv file based on Sampling Frequency, End Time, Window Parameter.

② Create Target Output File

a. .target file is a binary format for better performance.

b. After specifying the file name and the path, click **Create Target Output File** button to create .target.

c. Click **Plot** button to plot the data in .target.

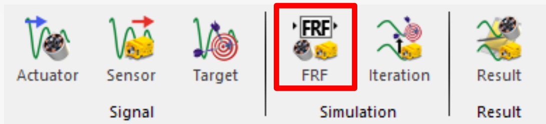


Tutorial (7)

1. FRF (1)

① FRF (Frequency Response Function)

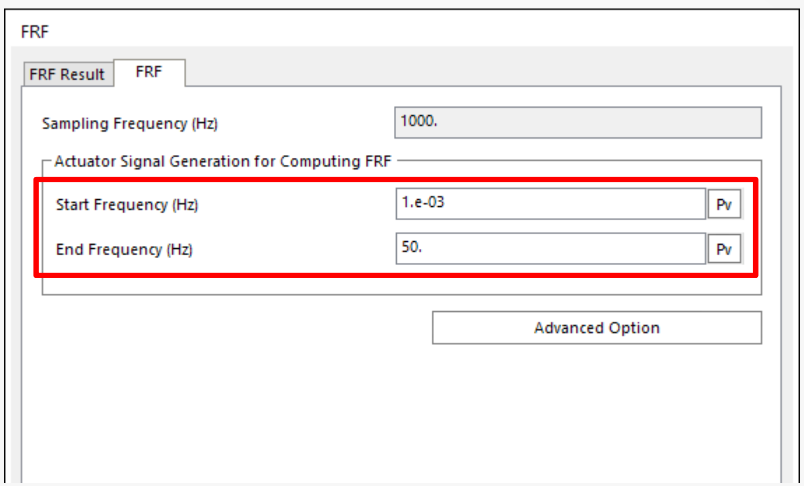
a. Computes the linearized model for System Identification (Transfer Function, H(f)).

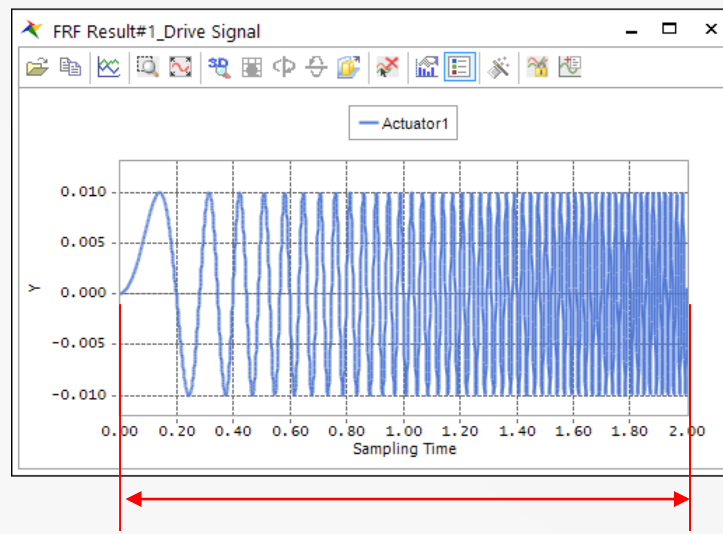


② Procedure (1)

a. Start/End Frequency(Hz)

- To perform FRF, the frequency of the signal for the actuators ('TACT(Actuator1)') is gradually increased using a Sweep Sine Function. Start/End Frequency define the sweep sine function.
- Since 0Hz is not valid, **Set Start Frequency** 0.001Hz.
- Since the Target Signal is the data below 50Hz, set **End Frequency** 50Hz.





RECURDYN

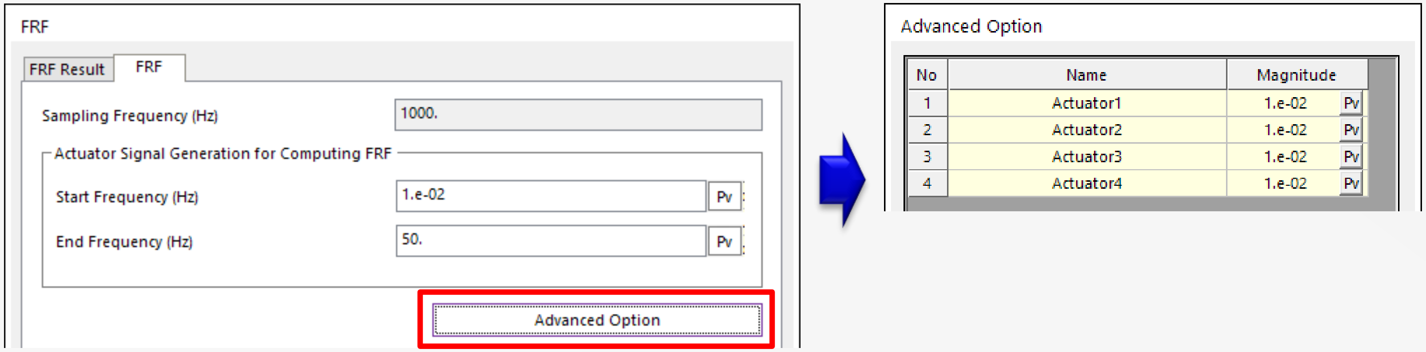
Tutorial (8)

1. FRF (2)

① Procedure (2)

a. Set Magnitude of Sweep Sine Function in **Advanced Option**.

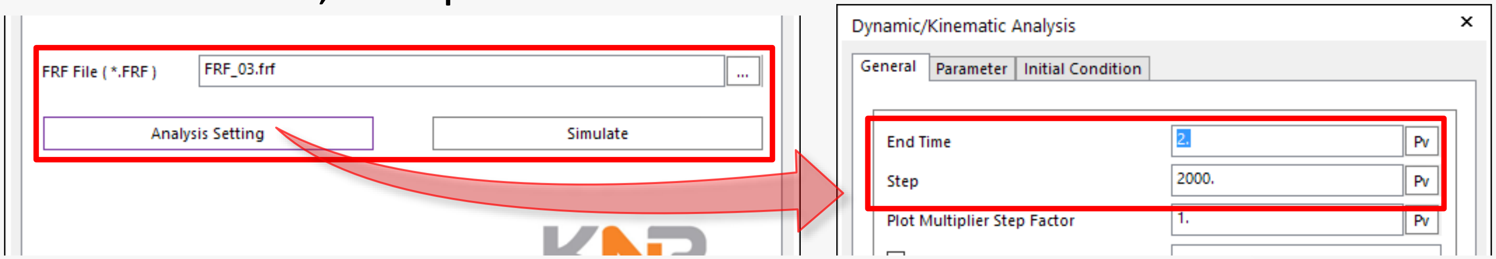
- Since the model in this tutorial uses MKS unit, Magnitude = 1 means that the displacement of the tire is 1m, which is too large for this model.
- Therefore set all the **Magnitudes** 0.01 to displace the tires by 10 mm.



b. Specify the file name and path for FRF result.

c. Adjust **Analysis Setting for** Dynamic Analysis, then click **Simulation** button.

- End Time and Step must be consistent with the Sampling Frequency.
- Since the Sampling Frequency in this tutorial is 1000Hz,
- Set **End Time = 2sec, and Step = 2000.**



Tutorial (9)

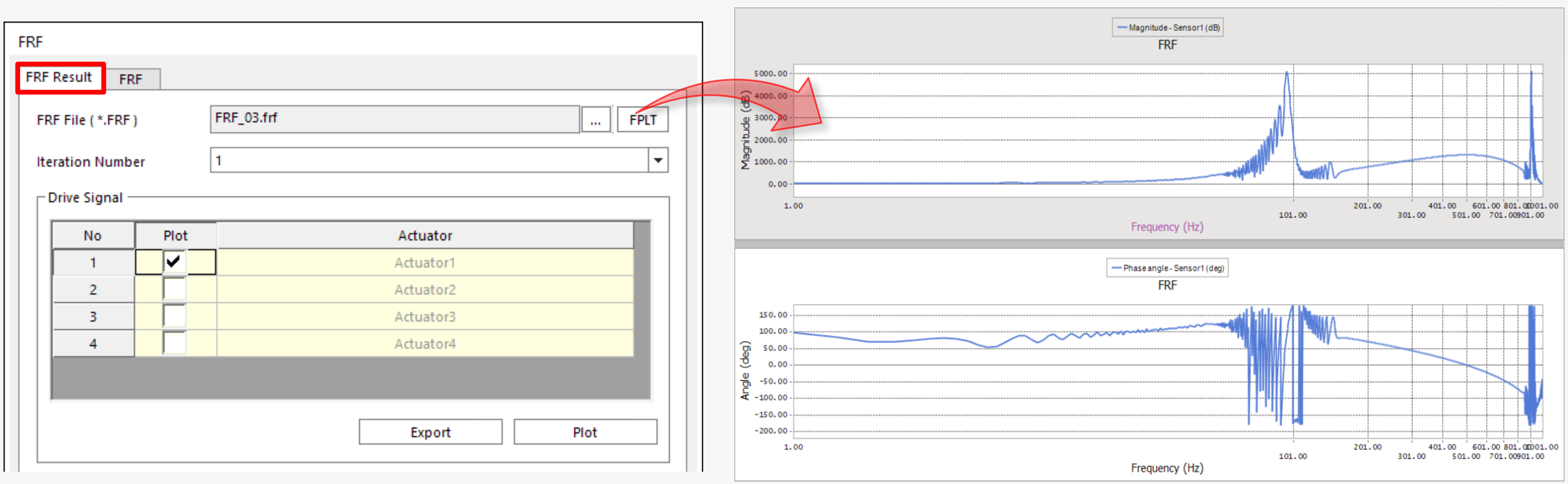
1. FRF (3)

① Procedure (3)

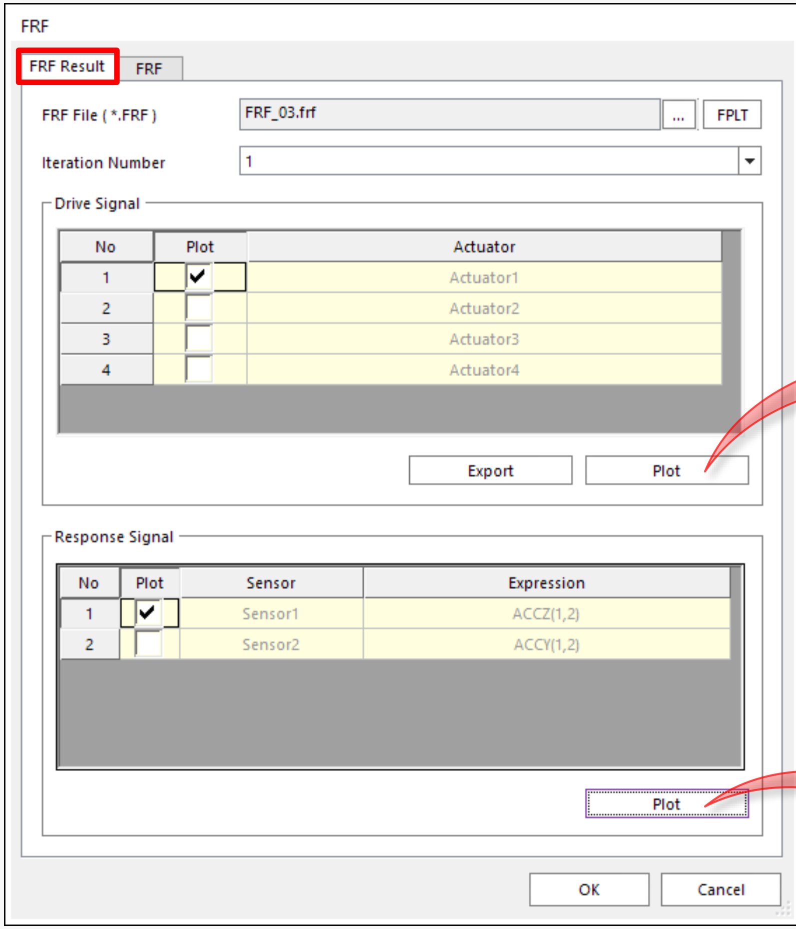
a. After you click **Simulation** button, the simulation is performed once for each actuator.
- In this tutorial 4 simulations are performed.
- During the FRF, the Sweep Sine Function is applied to one Actuator while all the other actuators are held at 0.

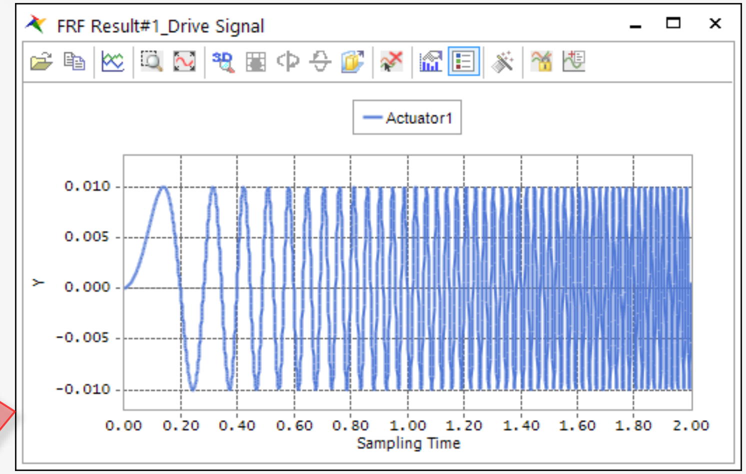
b. After simulation, under the FRF Result tab, you can click **FPLT** to plot the frequency responses of the system.

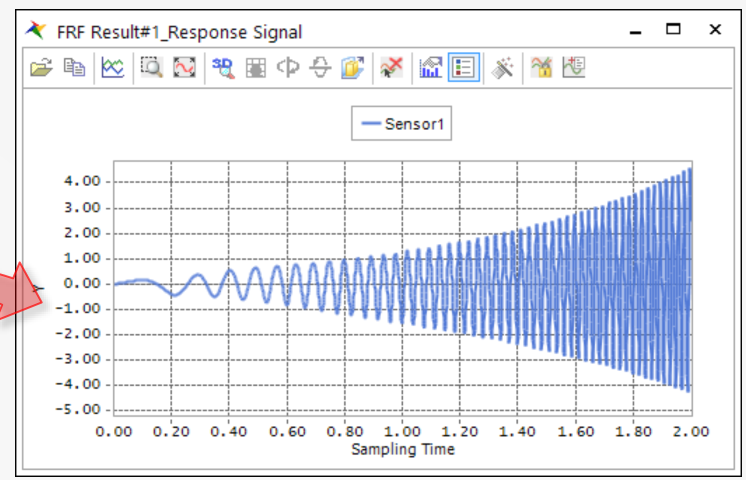
c. You can also **Plot** the **Drive Signal** (Sweep Sine Function) of the actuators and **Response Signal** of the sensors (next page).



Tutorial (10)


FRF
FRF Result
FRF
FRF File (*.FRF)
FRF_03.frf
...
FPLT
Iteration Number
1
Drive Signal
No
Plot
Actuator
1
Actuator1
2
Actuator2
3
Actuator3
4
Actuator4
Export
Plot
Response Signal
No
Plot
Sensor
Expression
1
Sensor1
ACCZ(1,2)
2
Sensor2
ACCY(1,2)
Plot
OK
Cancel

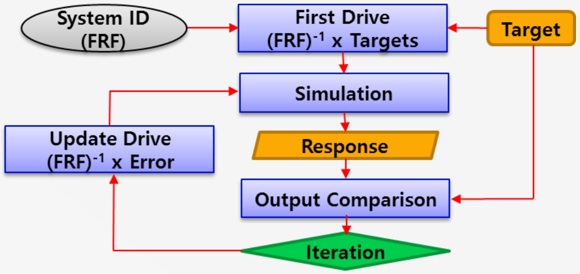

FRF Result#1_Drive Signal
Actuator1
0.010
0.005
0.000
-0.005
-0.010
0.00
0.20
0.40
0.60
0.80
1.00
1.20
1.40
1.60
1.80
2.00
Sampling Time

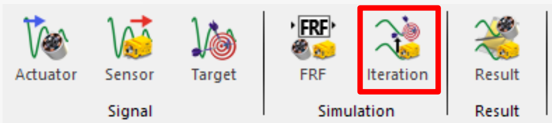

FRF Result#1_Response Signal
Sensor1
4.00
3.00
2.00
1.00
0.00
-1.00
-2.00
-3.00
-4.00
-5.00
0.00
0.20
0.40
0.60
0.80
1.00
1.20
1.40
1.60
1.80
2.00
Sampling Time

Tutorial (11)

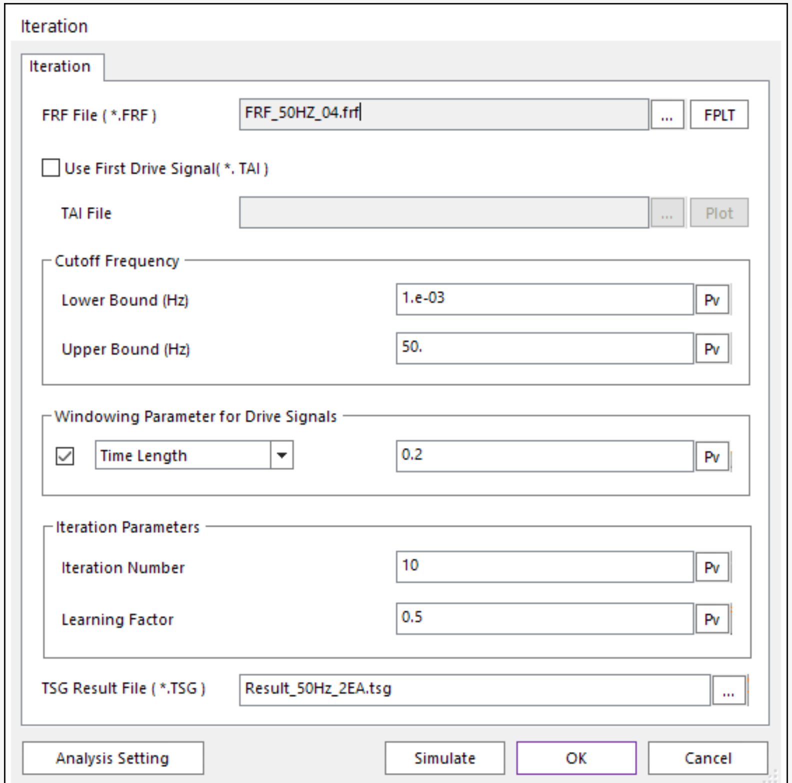
1. Iteration

① **Iteration**: performs **iterative simulations** to find the **Drive Signals** applied to Actuators to match the **Response Signal** of sensor to the **Target Signal** as closely as possible using FRF results.



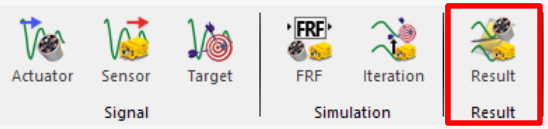


- **Procedure**
  - **FRF file:** FRF result calculated in the previous step is set automatically. Or you can specify .frf file.
  - **Cutoff Frequency and Windowing Parameters:** Use the same values used during FRF.
  - **Iteration Parameters:** Set Iteration Number = 10 and use the default value for Learning Factor (0.5).
  - **TSG Result File:** Specify the file name and path to save the TSG Results.
  - Click **Simulate** button.



Tutorial (12)

1. Result (1)



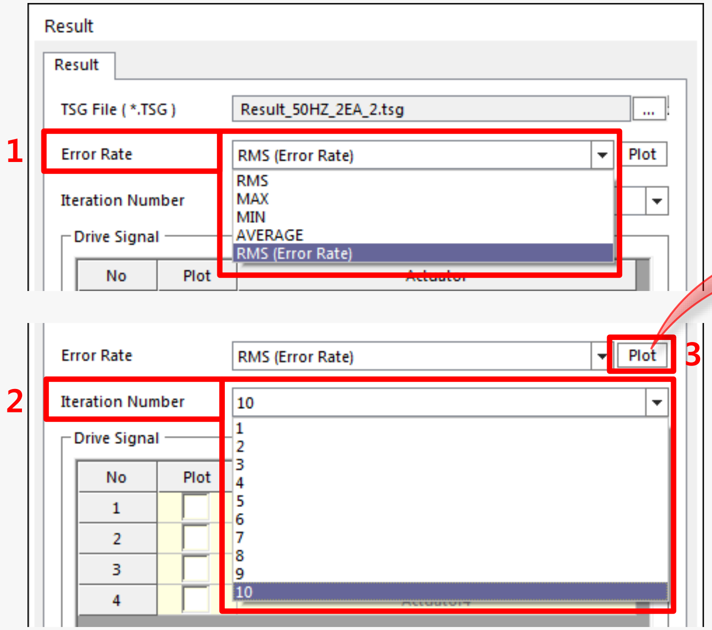
① Post-processor of TSG to review the result in .tsg after iterative simulations.

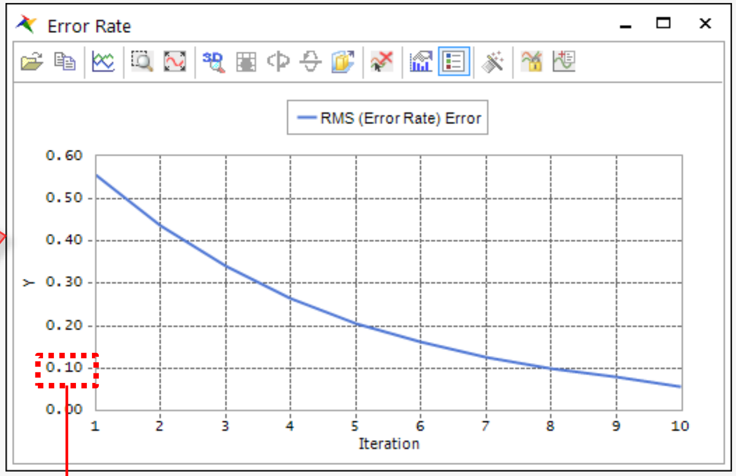
a. **Error Rate (RMS)**: For each iteration, the RMS of the difference between Response Signal (Sensor) and Target Signal at every instant.

b. **Error Rate (RMS (Error Rate))**: The relative difference between the RMS of Target Signal for entire time and the RMS of Response Signal (Sensor) for entire time.

② Procedure (1)

a. Specify the type of **Error Rate**, **Iteration Number**, and click **Plot** button to review the error rate of that iteration.





0.1 means there is 10% error between Target Signal and Response Signal

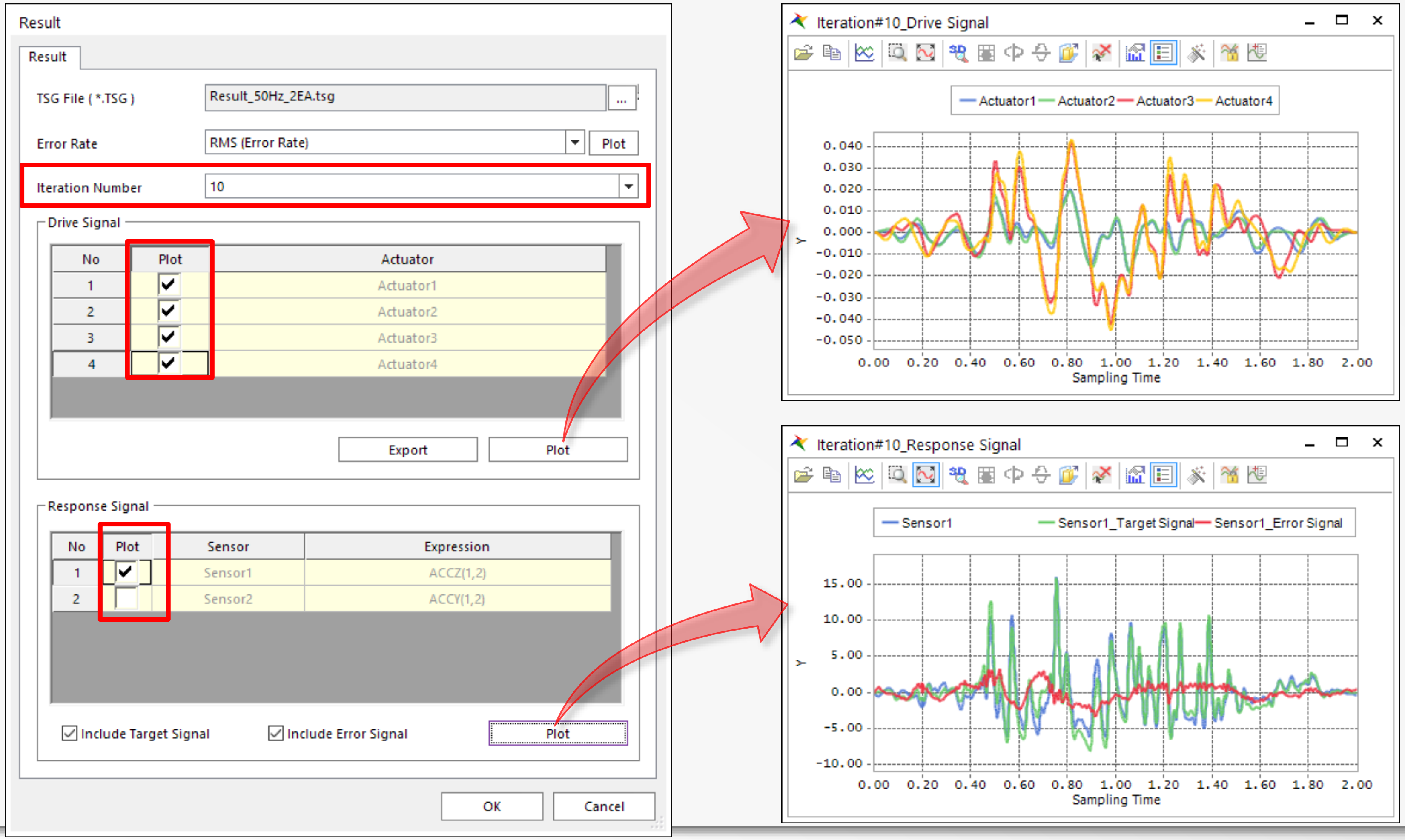
Tutorial (13)

1. Result (2)

① Procedure (2)

a. Select the desired **Iteration Number**

b. Plot the Drive Signal of the selected actuators and Response Signal of the selected Sensors.



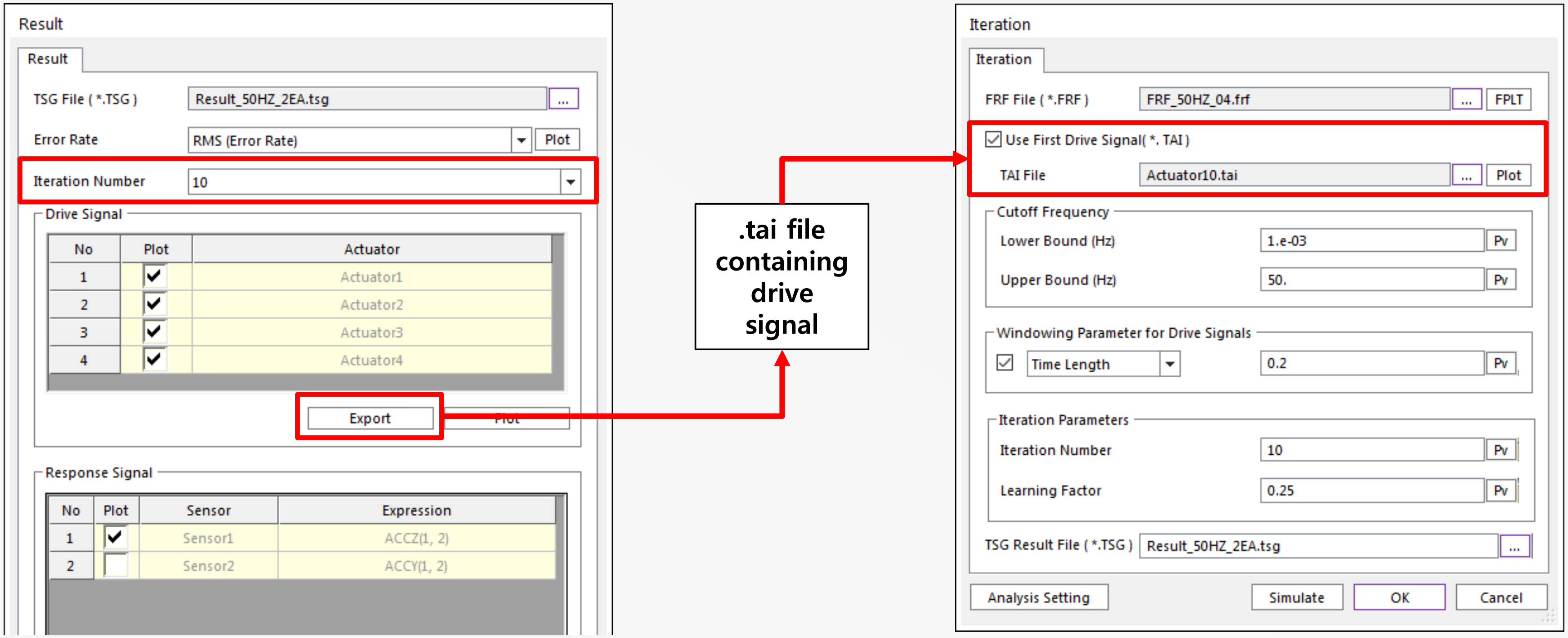
Tutorial (14)

1. Result (3)

① Procedure (3)

a. To perform additional iterations, drive signal results from the last iteration can be transferred to the next iterations using a .tai file.

- Select **Iteration Number** and click **Export**, storing Drive Signals of selected Iteration Number in .tai file.
- In Iteration dialog, check **Use First Drive Signal** and **specify** .tai file you just generated.



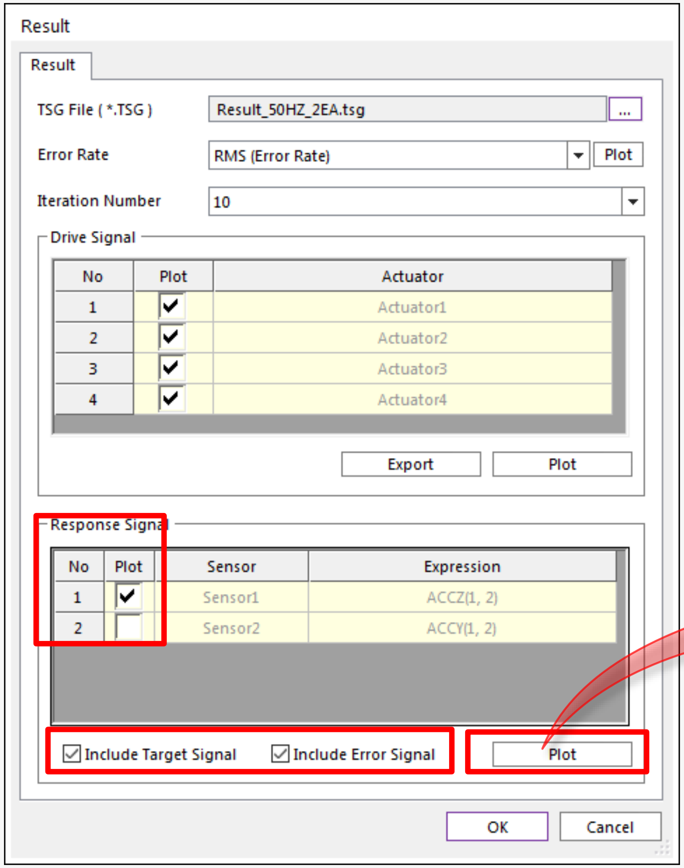
Tutorial (15)

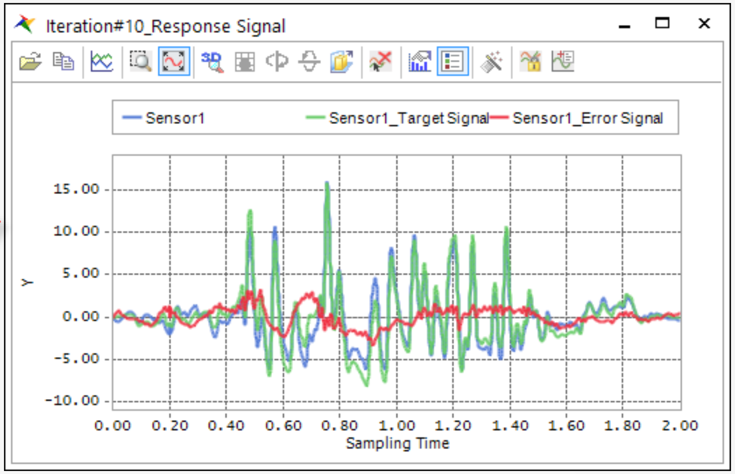
1. Result (4)

① Procedure (4)

a. When you plot Sensor data in ‘Response Signal’

- You can plot Target Signal or Error Signal as well as the output of Sensor.
- You can use the options **Include Target Signal** and **Include Error Signal**.





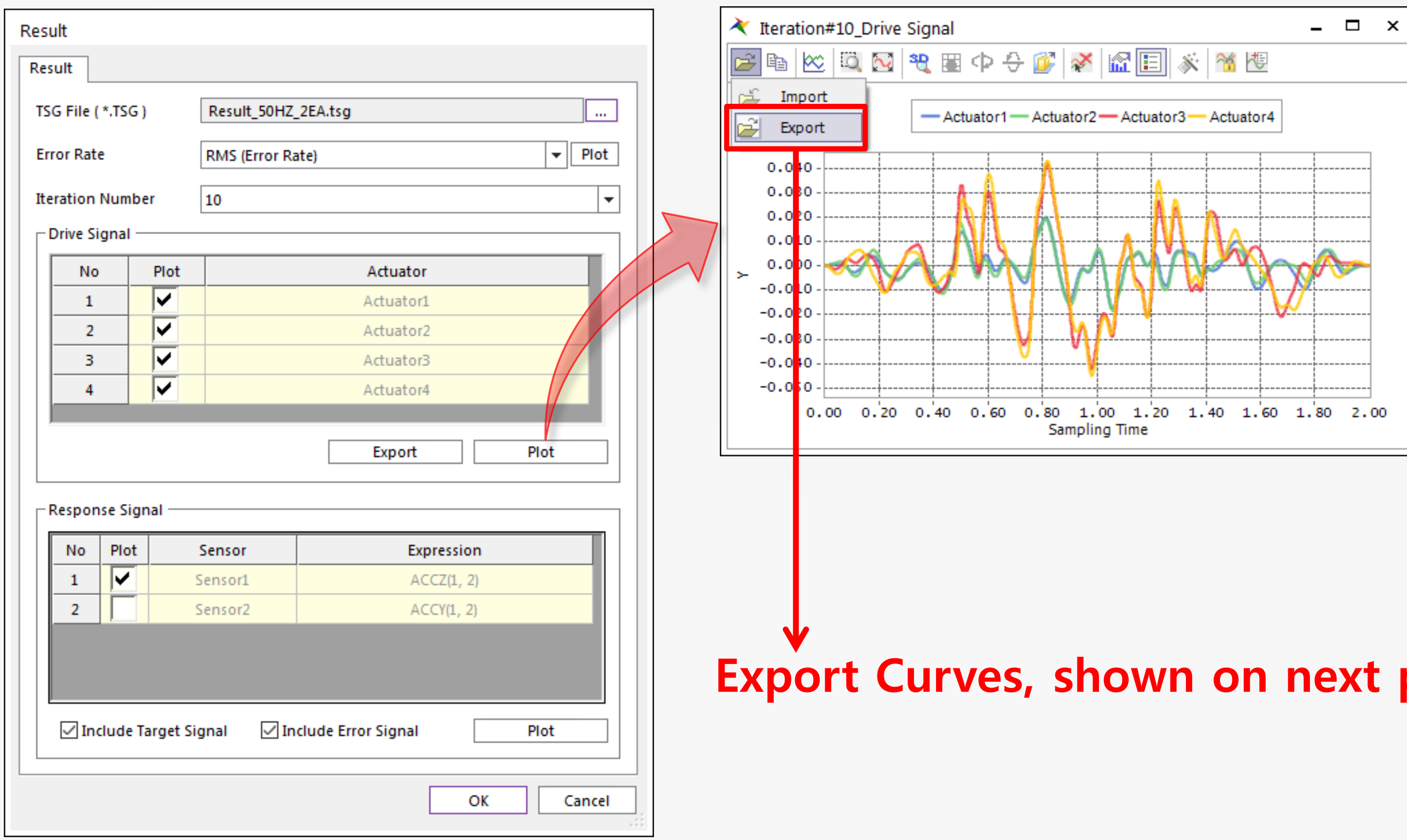
Tutorial (16)

1. Result (5)

① Procedure (5)

Once the user is satisfied with the drive signal results, they can transfer the data into a spreadsheet or text file to be used in other models.

a. From the plot of the drive signals, click the **Export** button.



Export Curves, shown on next page.

Tutorial (17)

1. Result (6)

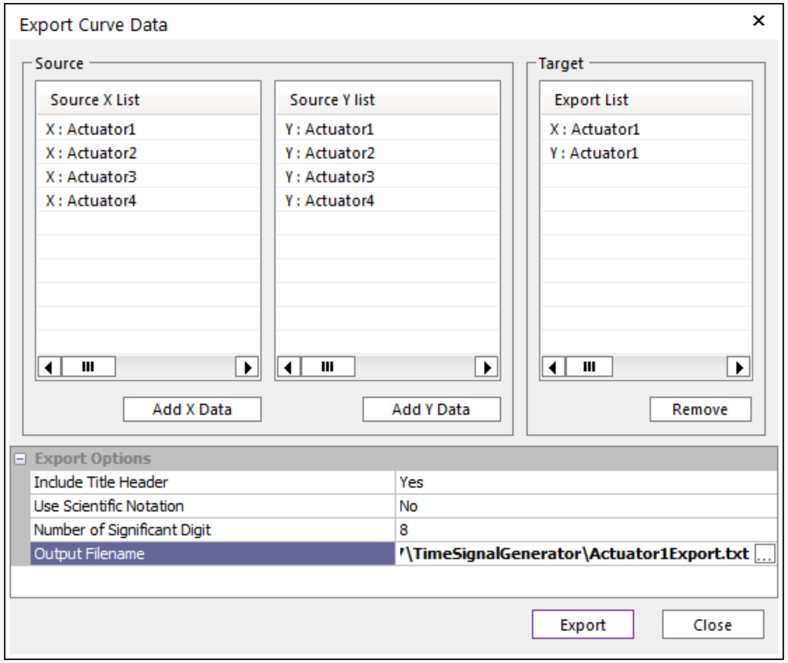
① Procedure (6)

From the Export Curve Data window that appears, the user can then export the data for each curve to a file, which can be imported into another model to be used as a driving signal.

a. Load the Export List with data from the Source X List and Source Y List.

b. Specify an Output Filename.

c. Click **Export**.



Conclusions

- RecurDyn/TSG (Time Signal Generator) allows analysts to utilize available physical test data to replicate loading conditions in an analogous MBD model.
- Provides a consistent, automated method for doing this.
- Removes the requirement for virtual terrain, tire, and driver modeling, which can be difficult to characterize correctly.
- Makes it easy for analysts to test new designs based on physical test data of existing designs.

Questions

Email support@motionport.com to obtain more information.