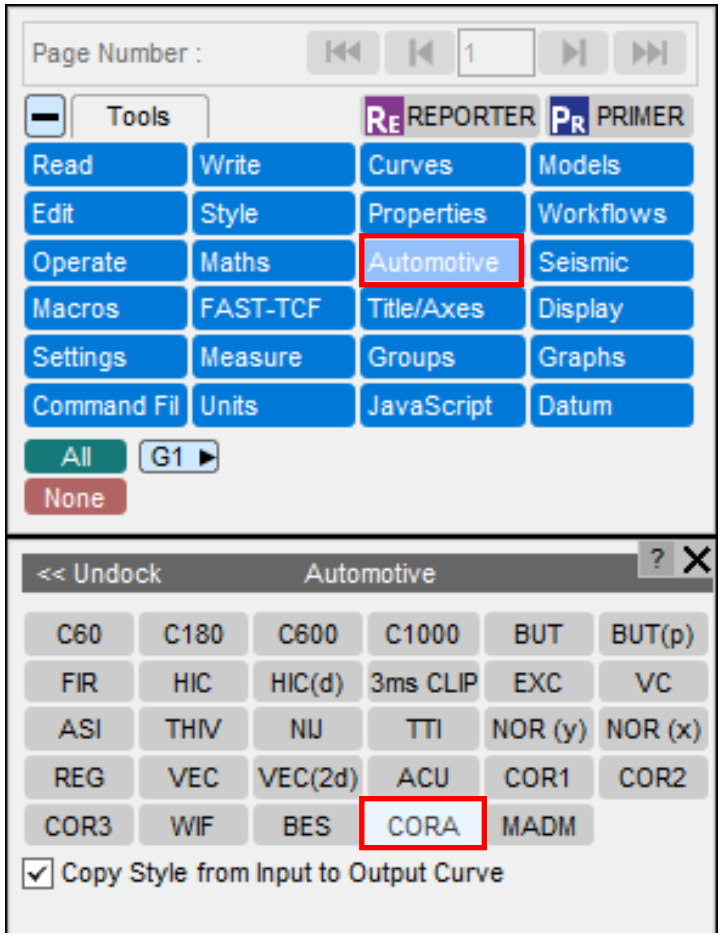
Curve correlation – CORAplus 4.0.4 and ISO/TS 18571:2014

Curve correlation

T/HIS provides a number of ways to determine a measure for the degree to which two curves match i.e. correlation.

Typically, one curve is a reference curve that might come from physical testing, the other is the comparison curve that may be the result of a simulation.

CORA Tool implements several curve correlation methods. This tutorial focuses on the **CORAplus 4.0.4** and **ISO/TS 18571:2014** methods, but it is applicable to the **ISO/TS 18571:2024** method too.

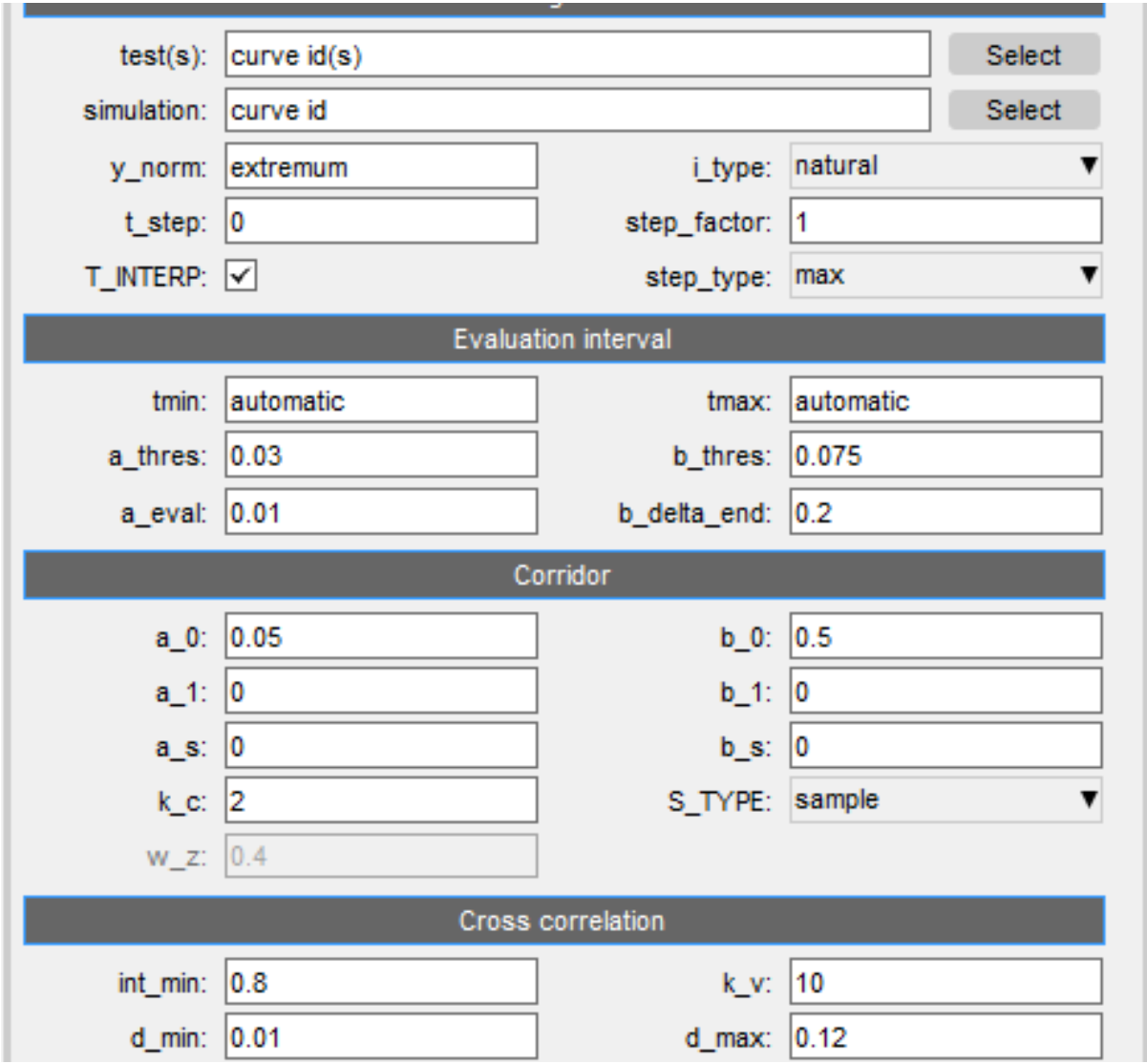


Assumptions and requirements

To achieve a meaningful correlation rating, the input curves should have a minimal amount of noise. This can be achieved using the filtering capabilities of T/HIS. It is assumed that any signal filtering required has been carried out prior to the use of CORA.

The curves should have x-values that increase monotonically.

The T/HIS CORA interactive tool currently assumes the curves for analysis are available for selection within T/HIS.



Example

Example – Introduction

The T/HIS CORA tool can be accessed via **Tools** → **Automotive** → **CORA**, or you can automate CORA calculations by importing the CORA JavaScript module into your own script.

This tutorial will demonstrate an example workflow using both the interactive tool, and the automated method.

The workflow demonstrated will be as follows:

1. Load experimental test data
2. Extract simulation results comparable to test data
3. Filter test and simulation time histories
4. Perform the CORAplus 4.0.4 and the ISO/TS 18571:2014 correlation

Note: it is assumed that you have some knowledge of curve reading and curve operations in T/HIS. As such, these processes are not described in any detail.

Example file package

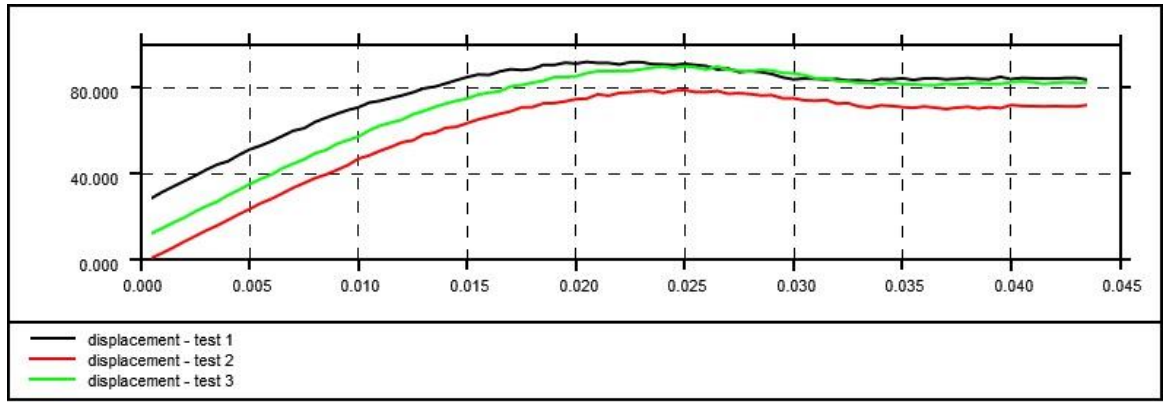
The example file package ($OA_INSTALL/manuals/tutorials/this/**cora_and_madm_example_files**) contains the following folders and files:

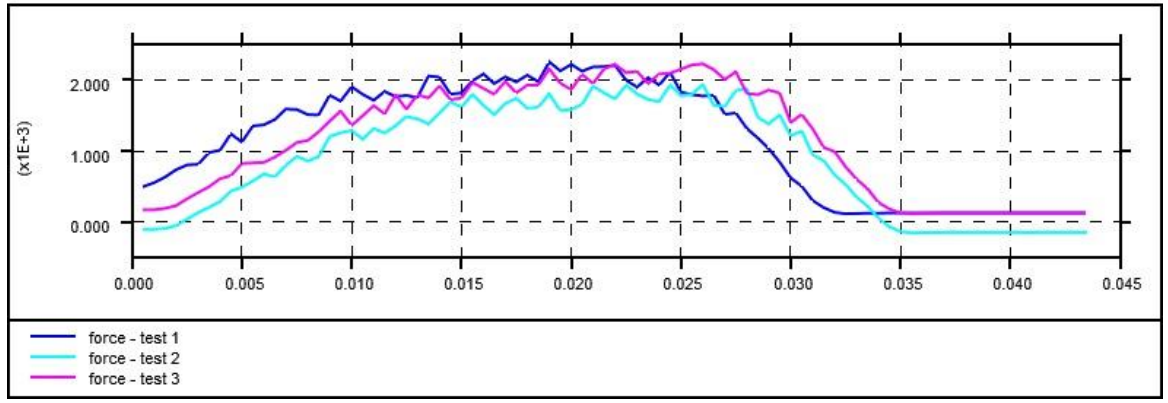
- The **analysis_model** folder contains an example LS-DYNA analysis model with results
- The **test_data** folder contains force and displacement time histories measured during testing
- The files ***cora_and_iso18571_workflow_test.js*** and ***cora_and_iso18571_workflow_test_input.json*** are the JavaScript and configuration file associated with this example.

```
.
├── analysis_model
│   ├── binout0000
│   ├── bouncer.key
│   ├── bouncer.ptf
│   ├── bouncer.ptf01
│   ├── bouncer.ptf02
│   ├── bouncer.thf
│   └── bouncer.ztf
│   └── bouncer.cur
├── test_data
│   └── bouncer.cur
├── cora_and_iso18571_workflow_test.js
└── cora_and_iso18571_workflow_test_input.json
```

1. Load experimental test data

The test data is read into T/HIS from the supplied curve file: **./test_data/bouncer.cur**



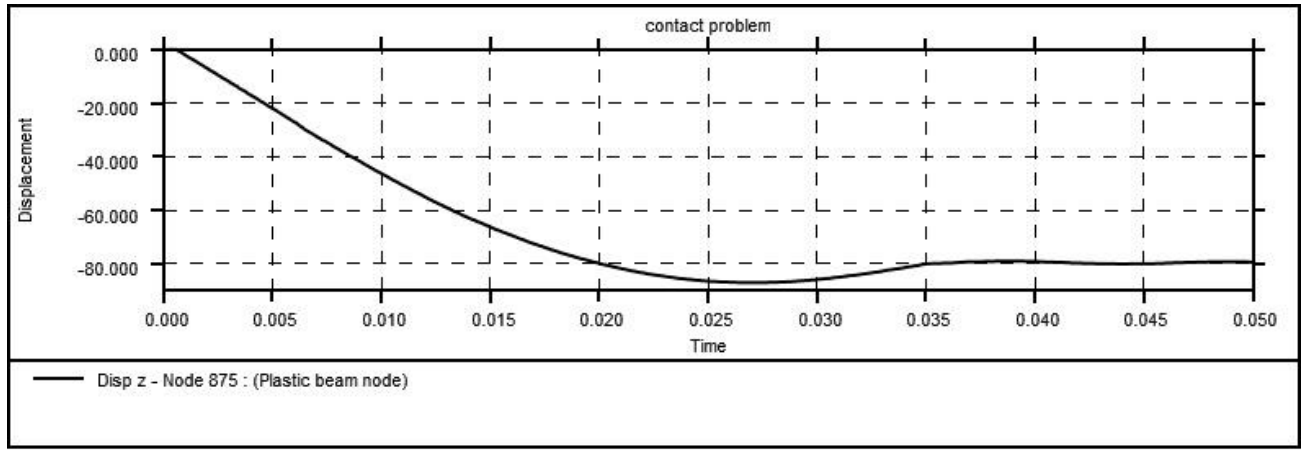


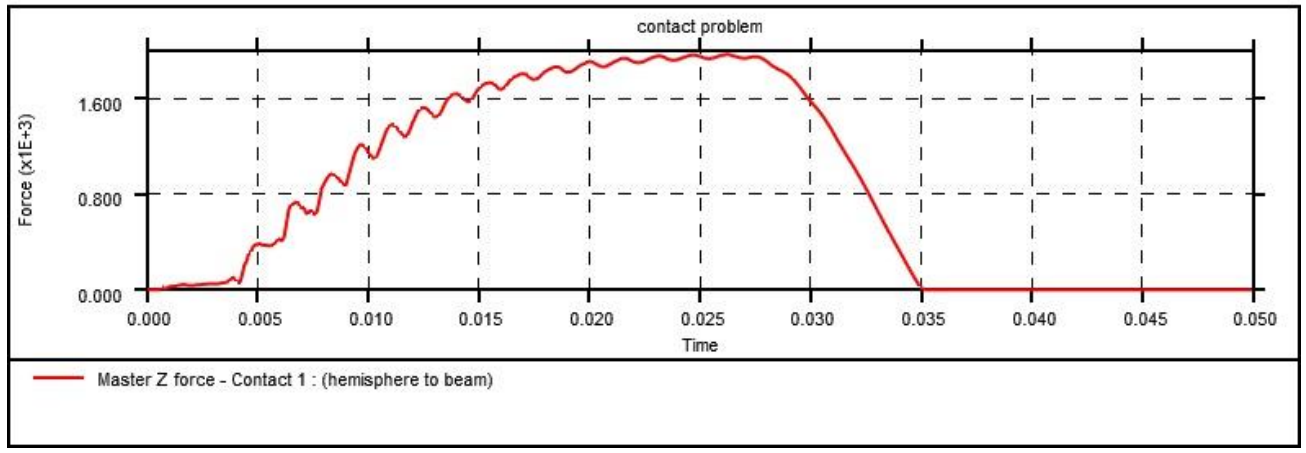

```
// T/HIS is able to import modules from paths that are relative to the
// T/HIS scripts library so the cora.mjs module can be imported as shown below.
// Alternatively, if you have changed you script library using oa_prefs you can
// use an absolute path and import from:
// "path/to/oasys/installation/this_library/scripts/CORA/modules/cora.mjs"
import { correlation_analysis, read_json } from "CORA/modules/cora.mjs";

// salient file paths as global variables
let script_dir = parent_directory(arguments[0]);
let model_file_path = script_dir + "/analysis_model/bouncer.thf";
let test_file_path = script_dir + "/test_data/bouncer.cur";
let cora_config_file = script_dir + "/cora_and_iso18571_workflow_test_input.json";

// read in the test curves - 3 tests - force and disp i.e. 6 curves total
let options = new Object(); // options for curve file read
let cur_id = Curve.FirstFreeID();
options.outputOpt = cur_id; // read in starting at first free id
Read.Cur(test_file_path, options);
// assign the curves to objects so can be manipulated
let cur_d_test = []; // array of test curves - displacement
let cur_f_test = []; // array of test curves - force
for (let i = 0; i < 3; i++) cur_d_test[i] = Curve.GetFromID(i + cur_id);
for (let i = 0; i < 3; i++) cur_f_test[i] = Curve.GetFromID(i + cur_id + 3);
```

Note: JavaScript functions are used to determine locations of the T/HIS executable and the directory from which the script is run. For simplicity, these locations could be hardcoded if desired.

2. Extract simulation results comparable to test data

The example model *.thf* file is read into T/HIS and the displacement and force curves are extracted:

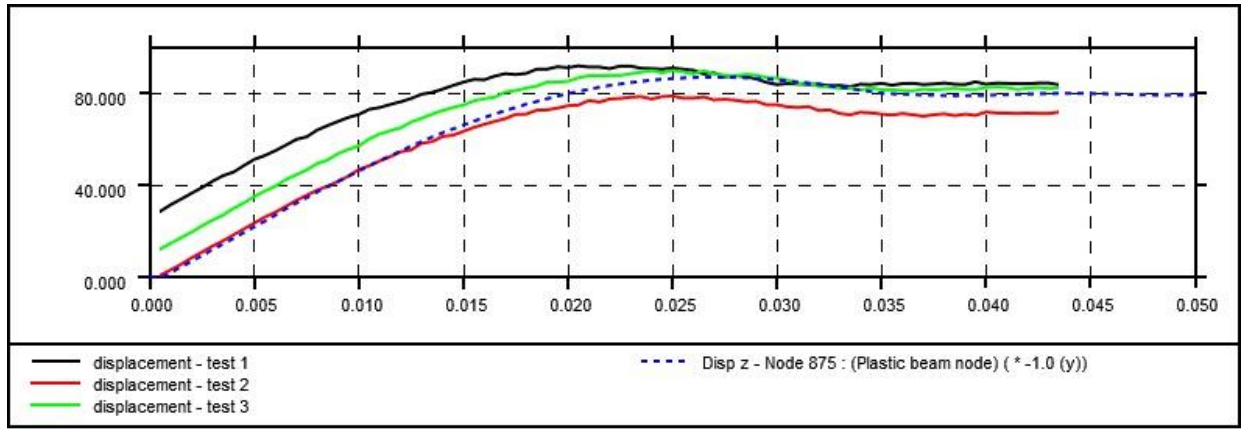


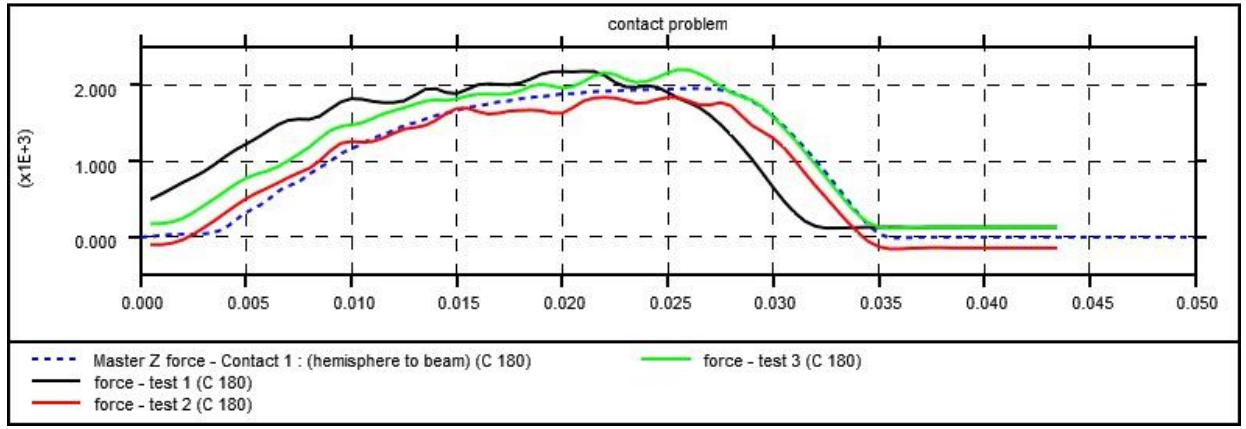



```
// read in the model
let m = Model.Read(model_file_path);

// first, extract displacement-time curve
let f_d = AllocateFlag();
m.SetFlag(f_d, Entity.NODE, -875);
let cur_d = m.GetDataFlagged(f_d, Component.DZ)[0];

// then, extract force-time curve
let f_f = AllocateFlag();
m.SetFlag(f_f, Entity.CONTACT, 1);
let cur_f = m.GetDataFlagged(f_f, Component.CFZ)[0];
```

3. Filter test and simulation time histories

The simulation displacement direction is reversed to match the test data and both simulation and test force data is passed through a C180 filter:





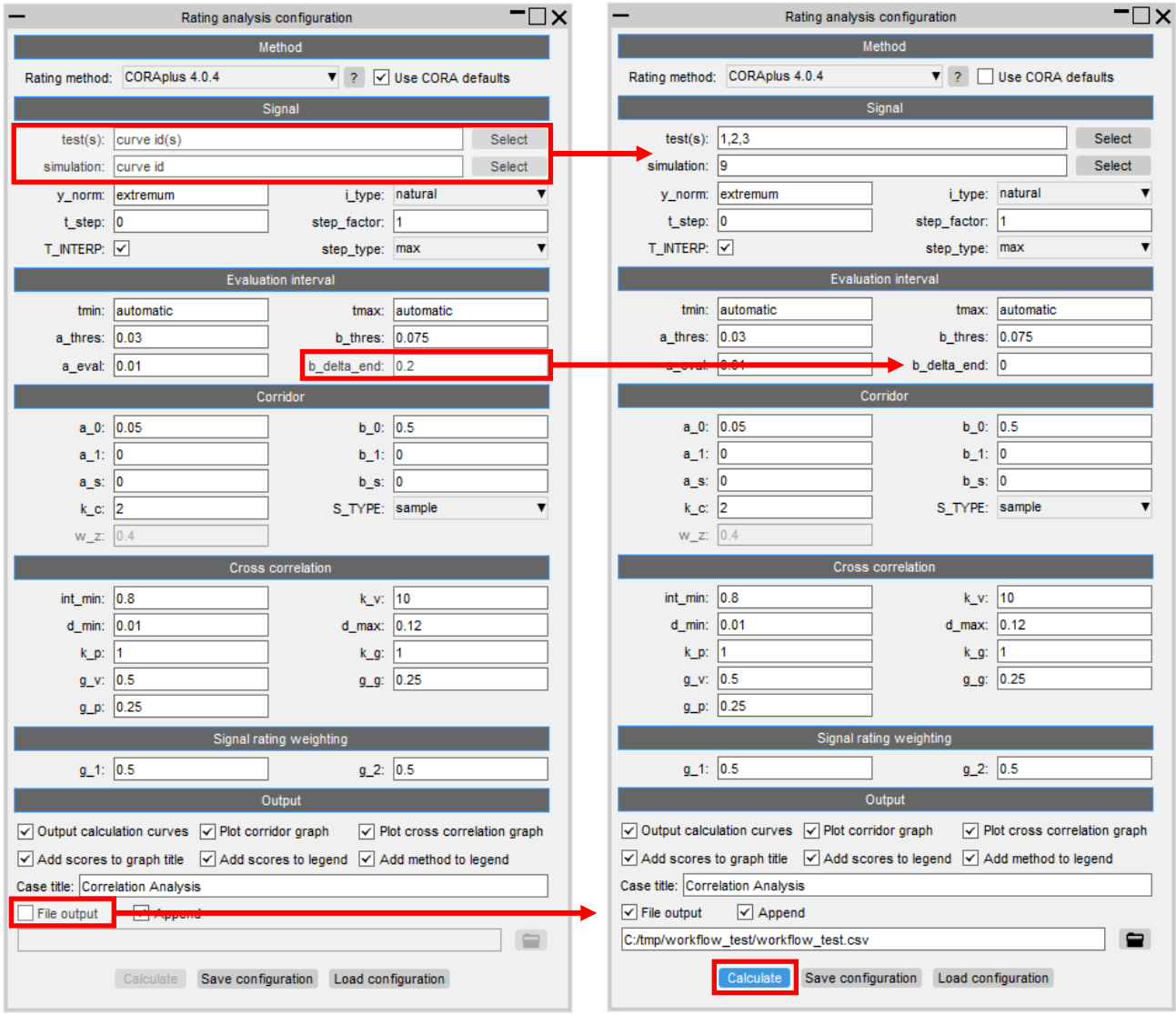

```
// flip displacement time
let cur_d_flipped = Operate.Mul(cur_d, -1);

// filter force-time
let cur_f_filtered = Operate.C180(cur_f)

// C180 filter the test force-time curves
let cur_f_test_filtered = [];
for (let i = 0; i < 3; i++)
    cur_f_test_filtered[i] = Operate.C180(cur_f_test[i])
```

4. Perform correlation calculations (interactive)

The CORAplus 4.0.4, ISO/TS 18571:2014 and ISO/TS 18571:2024 ratings can be obtained using the T/HIS CORA tool (**Tools → Automotive → CORA**).

Select the simulation and test curves, then alter the default CORA parameters as required. Here, b_delta_end = 0 as we have no oscillations at the end of the curve history.

We can also write the correlation analysis results to a file and output the calculation curves to T/HIS.

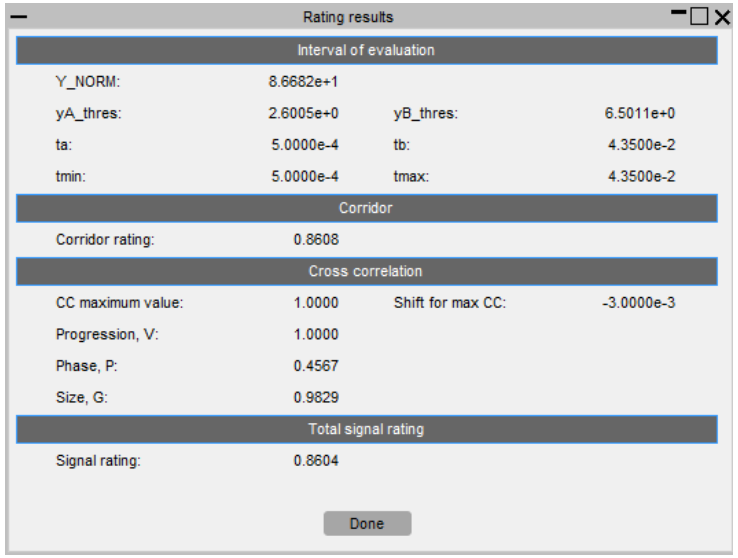


4. Perform correlation calculations (results)

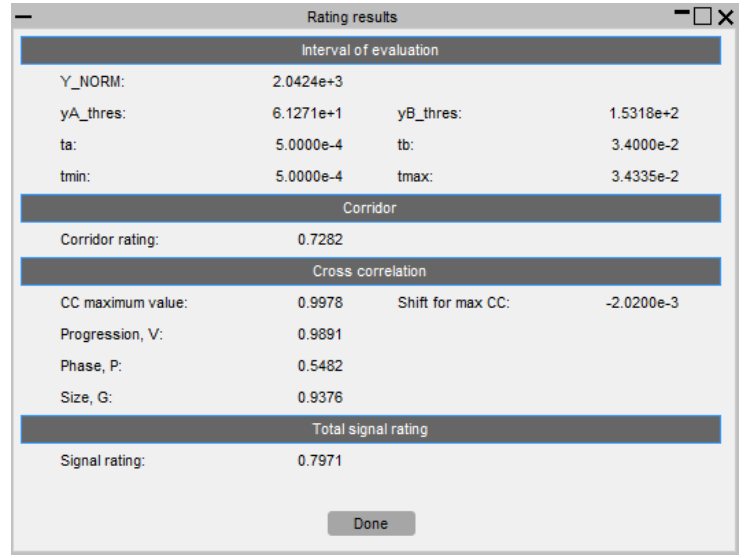
The correlation analysis is then performed by selecting the **Calculate** button. A results panel will then appear summarising the results of the correlation.

The following results are obtained for the displacement and force correlations for both the CORAplus 4.0.4 method and ISO/TS 18571:2014.

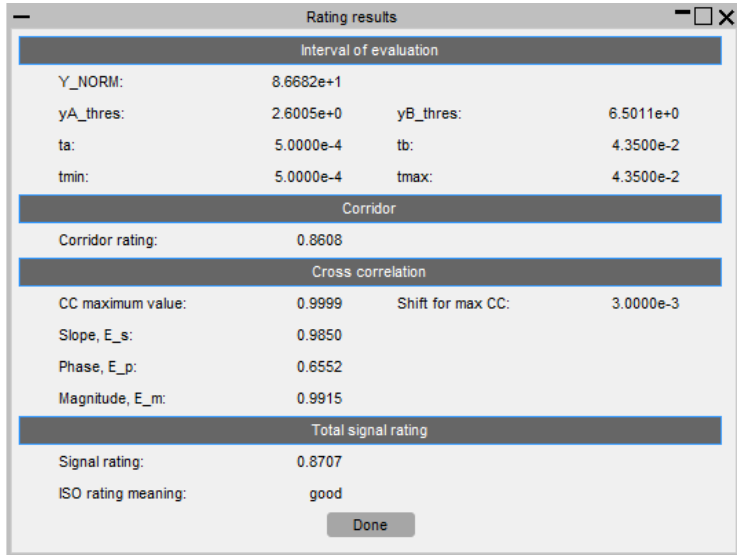
CORAplus 4.0.4 – displacement result



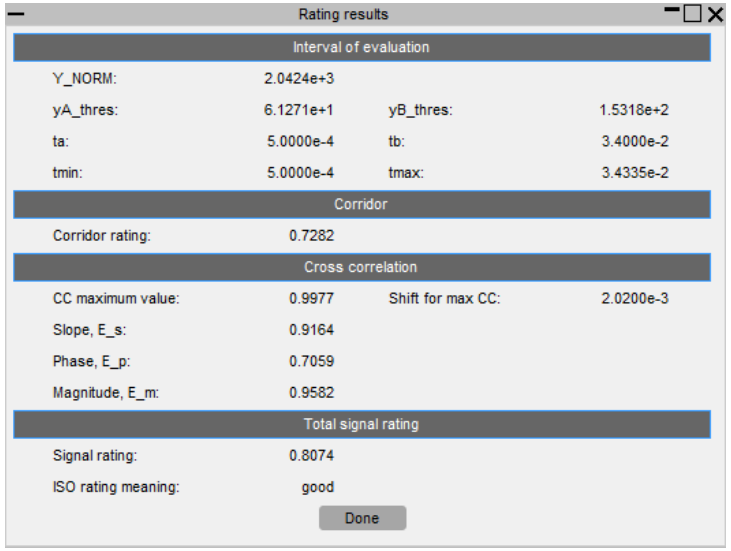
CORAplus 4.0.4 – force result



ISO/TS 18571:2014 – displacement result

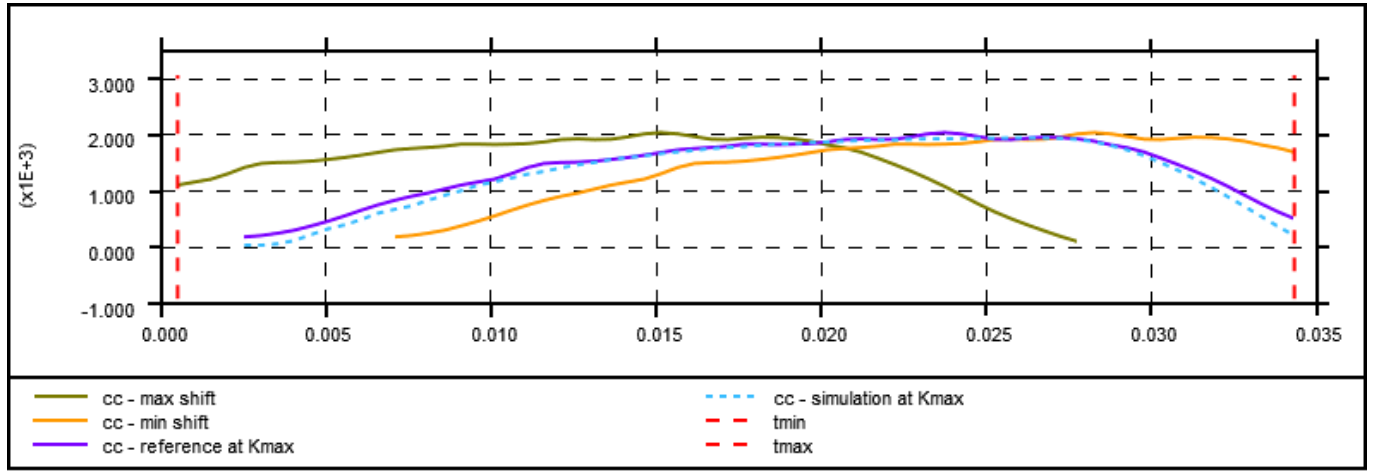


ISO/TS 18571:2014 – force result

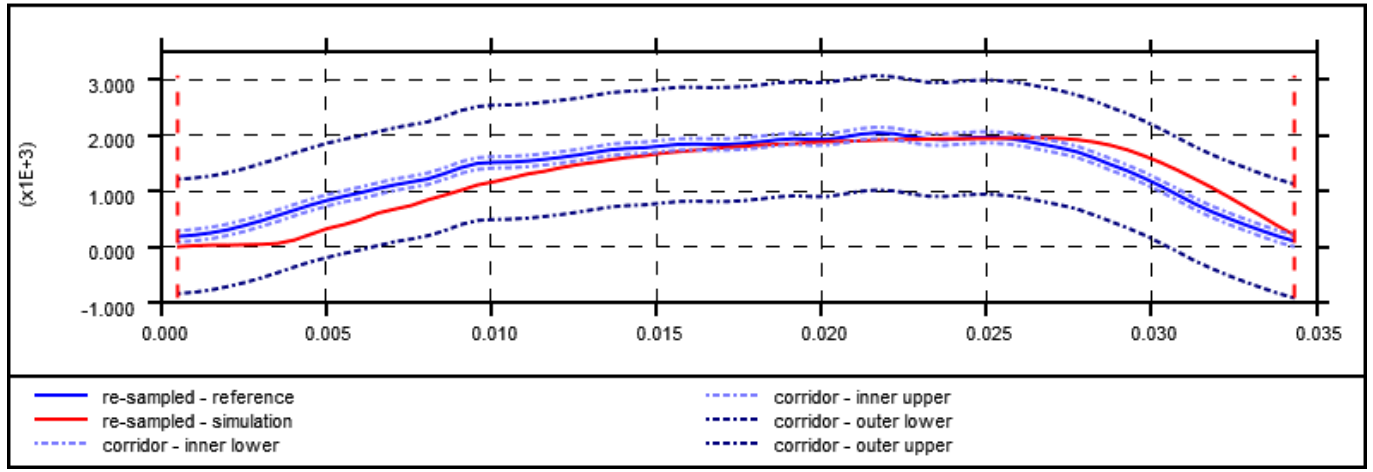


4. Perform correlation calculations (curve output)

A sample of the curve output, for the force correlation analysis, is shown below:



Curves calculated as part of the **cross correlation** component of the CORAplus 4.0.4 method analysis.



Curves calculated as part of the **corridor** component of the CORAplus 4.0.4 method analysis.

4. Perform correlation calculations (automated)

The correlation analysis can also be automated using JavaScript as shown below. See ***cora_and_iso18571_workflow_test.js*** in the example file package for a complete example.

```
function cora() {
    let cora_input = read_json(cora_config_file);

    if (cora_input == false) {
        ErrorMessage(`Error reading "${cora_config_file}" config file`);
        Exit();
    }

    /** overwrite file_output and b_delta_end property values
     * (other values can be overwritten in the same way)  */
    cora_input.file_output = script_dir + "/workflow_test.csv";
    cora_input.b_delta_end = 0.0;

    // turn on curve_output but only plot corridor (not cross-correlation) graphs
    cora_input.curve_output = true;
    cora_input.corridor_graph = true;
    cora_input.correlation_graph = false;

    // perform the pdb cora correlation analysis
    correlation_analysis(cora_input);

    // now update the properties to perform the ISO/TS 18571:2014 correlation analysis

    // first overwrite the method field
    cora_input.method = "ISO/TS 18571:2014";

    …
}
```

IMPORTANT: JavaScript Objects are case sensitive so ensure that all properties of cora_input are lower case or they will not be interpreted correctly

Configuration file and multiple cases

The ***cora_and_iso18571_workflow_test.js*** JavaScript uses a JSON format configuration file as input. All parameters that are configurable in the interactive tool (GUI) have a corresponding parameter. Both CORAplus 4.0.4 and ISO/TS 18571:2014 configuration parameters can be included in the same file.

In the example, the script explicitly sets two of these parameters (file_output and b_delta_end) after the configuration file has been read, i.e. further script-based configuration can be made if, for example, iterating on the sensitivity of different parameters.

In addition to the equivalent GUI parameters, it is also possible to define multiple cases so that more than one correlation analysis can be performed. For example:

```
"cases": [
    { "title": "cora - workflow test - displacement", "reference": [1, 2, 3], "simulation": 9 },
    { "title": "cora - workflow test - force", "reference": [11, 12, 13], "simulation": 10 }
]
```

In the example JavaScript, both the displacement and force correlation calculations are performed using the **CORAplus 4.0.4** method, before altering the correlation method to **ISO/TS 18571:2014** and performing another calculation.

When automating the correlation via JavaScript the results are written to the file_output CSV instead of being displayed in a pop-up window. For example:

run id	Y_NORM	yA_thres	yB_thres	ta	tb	tmin	tmax	corridor rating	max cc value	cc offset at max	progression rating	phase rating	size rating	Total signal rating	
cora - workflow test - displacement	86.68	2.60	6.50	0.0005	0.0435	0.0005	0.044	0.86	1.000	-0.003	1.00	0.46	0.98	0.86	
cora - workflow test - force	2042.36	61.27	153.18	0.0005	0.034	0.0005	0.034	0.73	0.998	-0.002	0.99	0.55	0.94	0.80	
cora - workflow test - displacement	86.68	2.60	6.50	0.0005	0.0435	0.0005	0.044	0.86	1.000	0.003	0.98	0.66	0.99	0.87	good
cora - workflow test - force	2042.36	61.27	153.18	0.0005	0.034	0.0005	0.034	0.73	0.998	0.002	0.92	0.71	0.96	0.81	good

Conclusion

This tutorial has demonstrated:

1. Loading experimental test data
2. Extracting simulation results comparable to test data
3. Filtering test and simulation time histories
4. Performing the CORAplus 4.0.4 and ISO/TS 18571:2014 correlation calculations and demonstrating the associated output

Contact us

Oasys
LS-DYNA Environment

Global / UK
T: +44 121 213 3399
E: dyna.support@arup.com

India
T: +91 40 69019723 / 98
E: india.support@arup.com

China
T: +86 21 3118 8875
E: china.support@arup.com

USA
T: +1 415 940 0959
E: us.support@arup.com

Subscribe to
our newsletter:

Follow us on:

@Oasys LS-DYNA Environment

@Oasys LS-DYNA Environment

@Oasys

@Oasys

www.oasys-software.com/dyna/