

CORApplus Release 4.1.0

User's Manual

Carsten Thunert, GNS mbH

Juni 2023

On behalf of PDB – Partnership for Dummy Technology and Biomechanics

GNS – Gesellschaft für Numerische Simulation mbH
Am Gaußberg 2
38114 Braunschweig
Germany
www.gns-mbh.de

pdb – Partnership for Dummy Technology and Biomechanics
Postfach 100 120
85001 Ingolstadt
Germany
www.pdb-org.com

Contents

1	Introduction	8
1.1	Corridor and cross correlation rating	8
1.2	Biofidelity rating according to ISO/TR 9790	9
1.3	Rating according to ISO18571	10
2	Getting started	11
2.1	Input Data	11
2.2	Results	11
2.3	Usage	12
3	Interval of evaluation	13
3.1	User-defined	13
3.2	Automatic calculation	13
3.3	Example of the calculation of the evaluation interval	14
4	Calculation of the total rating	17
4.1	Total signal rating	17
4.2	Load case rating	17
4.3	Sub-load case rating	18
4.4	Overall rating	18
5	Corridor method	19
5.1	Requirements	19
5.2	Determination of the corridors	19
5.2.1	Automatic corridors	20

5.2.2	User-defined corridors	21
5.3	Rating	21
5.4	Disadvantages	22
6	Cross correlation method	23
6.1	Requirements	23
6.2	Procedure	23
6.3	Maximum cross correlation	24
6.4	Progression rating V	25
6.5	Phase shift rating P	25
6.6	Size rating G	26
6.7	Rating	27
6.8	Example of the maximum time shift calculation	27
6.9	Disadvantages	28
7	Biofidelity rating according to ISO/TR 9790	29
7.1	Corridor method	29
7.2	Peak method	30
7.3	High/low peak method	30
7.4	Alternative high/low peak method	30
8	ISO18571 rating	32
8.1	Symbols and subscripts	32
8.2	The Corridor Rating	32
8.3	The cross-correlation Rating	33
8.4	Calculation of the overall ISO rating	36

9	Control file	37
9.1	Global parameters	38
9.1.1	Parameters to calculate the interval of evaluation	38
9.1.2	Parameters to format the HTML report	39
9.1.3	Small signals	40
9.1.4	Parameters to tune the corridor method	41
9.1.5	Parameters to tune the cross correlation method	42
9.1.6	Definition of the weighting factors by physical dimensions	43
9.1.7	Example of a global parameter block	43
9.2	Load case parameters	44
9.2.1	General Parameters	44
9.2.2	Data file block parameters	46
9.2.3	Signal block parameters	46
9.2.4	Example of a load case parameter block	50
9.3	Sub-load case parameters	51
9.3.1	General Parameters	51
9.3.2	Data file block parameters	51
9.3.3	Signal block parameters	51
9.3.4	Example of a load case block with two sub-load cases	51
10	Corridor files	53
10.1	Corridor method	53
10.2	Biofidelity method	53
10.2.1	Corridor and Peak rating	53
10.2.2	Peak low/high rating	53
11	Configuration file	55
11.1	Keywords	55
11.2	Obsolete Keywords	56

12 Definition of the system of units	58
12.1 Syntax	58
12.2 Basic units	58
12.3 Derived units	59
12.4 Units in the ASCII control file	59
12.5 Unit mapping file	60
13 Signal name mapping file	61
14 Program flow	62
15 CFC Filter	66
16 Spline interpolation	67
17 Encryption of the global parameters	69
17.1 Encryption	69
17.2 Decryption	69
17.3 Usage of encrypted parameters	70
17.4 Example	70
17.4.1 Encrypted parameter file	70
17.4.2 Control file	71
18 Proposed settings	72
18.1 Interval of evaluation	72
18.2 Corridor method	72
18.3 Cross correlation method	72
18.4 HTML report	73

19	References	74
20	Release notes	75
20.1	CORApplus 4.1.0	75
20.2	CORApplus 4.0.5	75
20.3	CORApplus 4.0.4	75
20.4	Cora 4.0.3	75
20.5	Cora 4.0.2	75
20.6	CORA 4.0	76
20.7	CORA 3.6.1	76
20.8	CORA 3.6	76

1 Introduction

CORApplus is a software that calculates the level of correlation between two non-ambiguous signals (e.g. time histories). The result ranges between “0” and “1” depending on the quality of match. So a rating of “1” represents a perfect match within defined tolerances and “0” with a poor match.

CORApplus is the acronym of **COR**relation and **Analysis**.

1.1 Corridor and cross correlation rating

CORA requires at least two curves to run an evaluation. The first curve is referred to as “reference curve” or $x(t)$, which can be a result of one test or the mean of several tests. The second curve is called “comparison curve” or $y(t)$, which can be a result of a numerical simulation or a test. The correlation between both is calculated within the time window $[t_{min}, t_{max}]$, also called the interval of evaluation.

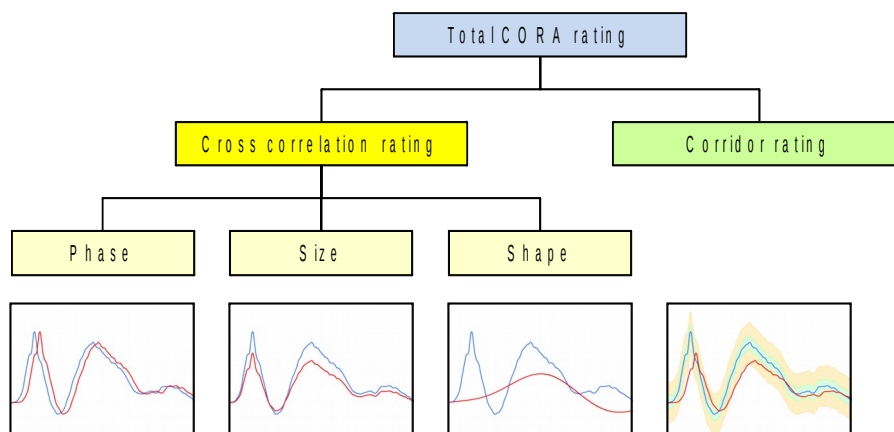


Figure 1: CORA rating structure

The correlation is calculated by using two different metrics – the corridor metric and the cross correlation metric. Both metrics are used to compensate each other’s disadvantages. Furthermore, the CORApplus rating tool is trying to separate engineer’s knowledge from the objective rating metric by using external parameters. So it is possible to fine-tune the evalua-

tion to the specific needs of the applications by adjusting those metric parameters to reflect the user's knowledge of the applications. The structure of the rating is shown in Figure 1.

The corridor method evaluates the deviation between two signals by means of corridor fitting. Four curves around the reference curve define the inner and the outer corridor. If the comparison curve is within the inner corridor, the result is "1". If the comparison curve is outside the outer corridor, the result is "0". Otherwise an interpolation of the rating is done. The corridor rating is applied to each time-step within the interval of evaluation. The total result of the corridor method is the mean of all single time-step ratings.

The cross correlation method analyses the characteristics of signals. It starts by moving the time signal of the reference curve by multiples of the time-step Δt within the interval of evaluation. For each of these time-shifts, the cross correlation value is calculated. At the time-shift of the maximum cross correlation, the three quantification values phase shift, size, and progression are calculated. The weighted sum of these three values is the total cross correlation rating.

The results of both metrics are finally summed up to the total CORApplus rating by using individual weighting factors for each metric.

1.2 Biofidelity rating according to ISO/TR 9790

The technical report ISO/TR 9790 [2] defines impact response requirements suitable for assessing the lateral impact biofidelity of the six body regions head, neck, shoulder, thorax, abdomen, and pelvis of crash test dummies, sub-component test devices, and mathematical models that are used to represent a 50th percentile adult male.

The rating is done for various responses by three different methods:

- Two curves define the upper and lower limit of a response (corridor rating)
- A lower and an upper limit specify the threshold for the peak of a response (peak rating)
- A lower or an upper limit specify the threshold for the peak of a response (peak low/high rating)

The result of a correlation can be "0", "5", or "10" points. The final rating is the weighted sum of the results of the six body regions. Several tests are described for each body region and

each test evaluates various responses. The following sliding scale is defined by ISO/TR 9790:

- Excellent biofidelity: $8.6 \leq rating \leq 10.0$
- Good biofidelity: $6.5 \leq rating < 8.6$
- Fair biofidelity: $4.4 \leq rating < 6.5$
- Marginal biofidelity: $2.6 \leq rating < 4.4$
- Unacceptable biofidelity: $0.0 \leq rating < 2.6$

The overall biofidelity rating of a side impact dummy or a mathematical model shall be greater than “2.6” to be acceptable for the evaluation of side impact occupant protection.

1.3 Rating according to ISO18571

The ISO standard 18571 74 provides validation metrics and rating procedures to be used to calculate the level of correlation between two non-ambiguous signals obtained from a physical test (TEST) and a computational model (CAE) and is aimed at vehicle safety applications.

The rating consists of two methods:

- A corridor rating which is the Cora Corridor method with a fix set of parameters.
- A cross-correlation rating which consists of three subratings as the Cora cross-correlation. But the subratings for phase, magnitude and slope correlation differ from the Cora cross-correlation ratings.

2 Getting started

2.1 Input Data

CORApplus is a command line software that is managed by an ASCII control file. The signal files, its structure (load cases and sub-load cases), and all parameters to select and tune the different metrics are specified by this file. If some parameters in the control file are missing, the user is asked to specify them in the command line directly. CORApplus only checks the parameters for a valid domain but not if the values make sense. The syntax of the control file is described in section 9.

The supported curve file formats are ISO/TS 13499 (ISO-MME) [3], Diadem, HyperGraph ASCII, LS-PrePost ASCII and PAM-VIEW ASCII. All curves are identified by a description which have to comply with the ISO/TS 13499 format [3]. It is possible to define a name mapping file to use different curve names (section 13).

CORApplus is available for Linux and MS-Windows computers. On Linux systems CORApplus shall be started by the script `CORApplus.sh`. This script defines an environment variable with the path of the software's installation (*[INST_PATH](#)*). On MS-Windows computers CORApplus shall be started directly or by using the batch file `CORApplus.bat`.

2.2 Results

The result of a CORApplus evaluation is a HTML report saved in the sub-directory "html" of the current working directory. The rating of each signal (curve) is presented on one page, respectively.

In general, the signals can be grouped in two or three levels:

- signal → load case → total rating
- signal → sub-load case → load case → total rating

The results of all signals belonging to a load case (LC) or a sub-load case (SLC) are summarised on a load case or sub-load case HTML page. The results of all load cases are listed

in the document `result_CORApplus.html`, located in the “html” directory. So it is recommended to start the review of an evaluation with this document.

The charts (images) shown on the signal pages of the HTML report are stored in the directory “images”. The HTML report refers to this directory, meaning if the report is moved to another location, the “images” folder must be moved too.

CORApplus uses HTML templates with predefined variables to create the HTML reports.

CORApplus creates the directory “curves” to save ASCII curve files. It is located in the working directory too. Each curve file contains the curves of one chart of a signal page. The format is selected by the keyword **OUTPUT_FORMAT** in the control file.

Apart from the HTML files, the ASCII file `result_CORApplus.txt` is written. It lists all input parameters as well as all results. CORApplus writes progress information in the file `CORApplus.log`. The file `CORApplus.msg` contains warnings and error messages. All of those files are located in the current working directory.

2.3 Usage

The software CORApplus shall be started from the command line. The first parameter has to be the control file followed by some optional command line parameters.

Linux
`CORApplus.sh <control file> [command line parameters]`

MS-Windows
`CORApplus.bat <control file> [command line parameters]`

CORApplus accepts the following command line parameters:

- a creates an Animator4 session file to read the curves and create the report (not officially supported feature)
- b the browser (HTML viewer) is started at the end of the evaluation (e.g. `result_CORApplus.html`)
- h name of the HTML main page (default: `result_CORApplus.html`)
- I set the path where the configuration files are found

o name of the ASCII result file (default: result_CORApplus.txt)

p no HTML output

3 Interval of evaluation

The interval of evaluation interval $[t_{min}, t_{max}]$ defines the domain of the x-axis (generally time) which is used to calculate the rating. The parameters $[t_{min}, t_{max}]$ use the system of unit specified in the file output_units.txt. The only supported units are [s] (seconds) and [ms] (milliseconds).

The interval of evaluation can be defined for each signal separately. However, its also possible to use the terms “x”, “y”, or “automatic” to enable specific functions. “x” can only be used locally. It means that the globally defined value is used. The global values for t_{min} and t_{max} are defined in the global parameter block by the keyword **T_MIN/T_MAX**. If “y” is used, the interval of evaluation of the first signal of a load case or sub-load case is used for all remaining signals of the same load case or sub-load case. It is a local setting too. The automatically calculated interval of evaluation is enabled by the keyword “automatic”. This setting can be used globally and locally but is not available for the biofidelity rating according to ISO/TR 9790.

3.1 User-defined

The user can specify the starting and ending time signal-wise or globally. It is also possible to define the starting time manually and to use the automatism to calculate the ending time (or the other way round).

3.2 Automatic calculation

The automatic calculation of the interval of evaluation is controlled by the global keywords **Y_NORM** (Y_{norm}), **A_THRES** (a_{thres}), **B_THRES** (b_{thres}), **B_DELTA_END** (b_{delta_end}), and **A_EVAL** (α_{eval}). The starting time is calculated by $y_{thresA} = a_{thres} \cdot Y_{norm}$ with $0 < a_{thres} \leq 1$. It is possible to set **Y_NORM** to “extremum”. In this case, Y_{norm} is calculated as the absolute maximum of the reference curve (section 5.2.1). The ending time of the interval is calculated by $y_{B_thres} = b_{thres} \cdot Y_{norm}$ with $0 < b_{thres} \leq 1$.

The starting time t_a is the first point in time when $|x(t)| > y_{A_thres}$ is true. The end of the evaluation interval t_b is the last point when $|x(t)| > y_{B_thres}$ is true. Furthermore, the time interval can be expanded by using the parameter α_{eval} .

As the bounds of the interval cannot exceed the domain of the reference curve $x(t)$, the interval is shorten automatically, if necessary. This leads to the final equation for t_{min} and t_{max} :

$$t_{min} = \max(t_a - \alpha_{eval} \cdot (t_b - t_a), t_{start\ x(t)}) \quad 0 \leq \alpha_{eval} \leq 1$$

$$t_{max} = \min(t_a + \alpha_{eval} \cdot (t_b - t_a), t_{end\ x(t)}) \quad 0 \leq \alpha_{eval} \leq 1$$

To shorten the interval of evaluation for curves with an almost horizontal but oscillating ending, the parameter b_{delta_end} (**B_DELTA_END**) is introduced. It defines a corridor around the end of the reference curve ($Y_{cor} = b_{delta_end} \cdot Y_{norm}$). The ending time t_{max} is reduced until the amplitudes of the reference curve vary by no more than Y_{cor} . This algorithm can be disabled by setting **B_DELTA_END** to "0".

If one of the calculated interval bounds does not match with a supporting point of the curve, reference curve and comparison curve are re-sampled to get supporting points at t_{min} and t_{max} .

At the very end, the determined interval of evaluation is checked against the domain of the comparison curve $y(t)$. If the interval is not fully covered by $y(t)$, it will be shortened to fit in the domain of $y(t)$.

3.3 Example of the calculation of the evaluation interval

The functioning of **B_DELTA_END** is described in this section exemplary by using the signal shown in Figure 2. The corresponding values of the curve are shown in Table 1.

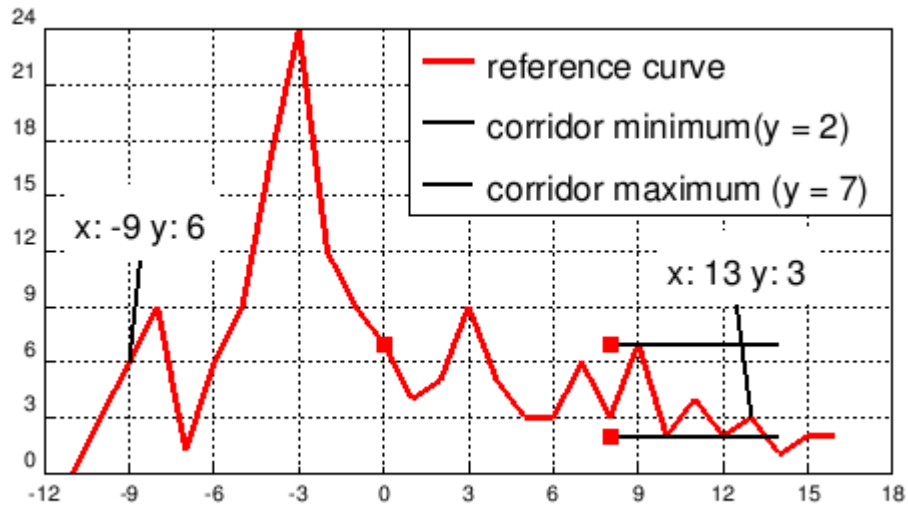
Figure 2: Example of B_DELTA_END

Table 1: xy values of the reference curve

-11	-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
0	3	6	9	12	6	9	17	24	12	9	7	4	5	9	5	3	3	6	3	7	2	4	2	3	1	2	2

The following global parameters are defined:

```

A_THRES    0.24
B_THRES    0.125
A_EVAL     0.1
B_DELTA_END      0.2
T_MIN/T_MAX      automatic  automatic
Y_NORM      extremum

```

As Y_NORM is set to “extremum”, the curve maximum $y=24$ at $x=-3$ is used to calculate $y_{A_thres} = a_{thres} \cdot Y_{norm} = 0.25 \cdot 24 = 6$ and $y_{B_thres} = b_{thres} \cdot Y_{norm} = 0.125 \cdot 24 = 3.0$. So the interval starts when the amplitude exceeds “6” and ends when the signal falls below “3”.

The requirements to start the interval are met at $t_a = -9$. At $t_b = 13$ the requirements to end the interval are met. Thus, the resulting interval length is $t_b - t_a = 13 - (-9) = 22$.

The option B_DELTA_END is enabled in this example and it is set to “0.2”. The corresponding corridor width Y_{cor} is calculated:

$$Y_{cor} = b_{\text{delta_end}} \cdot Y_{norm} = 0.2 \cdot 24 = 4.8$$

CORApplus starts the iteration at $t_{max} = 13$ and is going backwards along the x-axis. The corridor is made of the y values of the analysed time domain $t_{max} - n \cdot \Delta t$ and updated at every iteration step. The iteration stops when the maximum corridor width Y_{cor} is reached ("4.8" in this example). So the new end of the interval is now set to $t_b = 9$. The iterations of this example are shown in Table 2.

Table 2: Iterations to calculate the end of the interval of evaluation

X	Y	Corridor Lower bound	Corridor Upper bound	Corridor Width	Corridor exceeds limit of 4.8?
13	3	3	3	0	no
12	2	2	3	1	no
11	4	2	4	2	no
10	2	2	4	2	no
9	7	2	7	5	yes

The last modification of the interval is done by the parameter α_{eval} :

$$t_{min} = t_a - \alpha_{eval} (t_b - t_a) = -9 - 0.1 \cdot 22 = -9 - 2.2 = -11.2$$

$$t_{max} = t_b + \alpha_{eval} (t_b - t_a) = 9 + 0.1 \cdot 22 = 9 + 2.2 = 11.2$$

The first supporting point is "-11". Therefore, the curve is re-sampled to get a supporting point at "-11.2". t_{min} is set to "-11.2".

In summary, the automatically calculated interval of evaluation starts at $t_{min} = -11.2$ and ends at $t_{max} = 11.2$.

4 Calculation of the total rating

Primarily, CORApplus analyses the correlation of two signals and calculates a rating score depending on the goodness of correlation. However, the software calculates average ratings of the analysed signals too. The number and structure of those average ratings depends on the organisation of the signals in the control file.

Section 9.2.3 and 9.3.3 give information on the weighting factors of signals, section 9.3.1 sub-load cases, and section 9.2.1 of load cases. The weighting factors of signals, sub-load cases and load cases are normalised. The global parameter **WF_NORM** can be set to “NO” to show non-normalised weighting factors in the HTML report.

4.1 Total signal rating

The total signal rating C_3 of CORApplus is the sum of the weighted ratings of the corridor metric C_1 and the cross correlation metric C_2 .

$$\text{Signal rating} \quad C_3 = g_1 \cdot C_1 + g_2 \cdot C_2$$

The weighting factors g_1 (keyword **G_1** and parameter **g_1**) and g_2 (**G_2**, **g_2**) must comply with the following conditions:

$$\text{Corridor rating} \quad 0 \leq g_1 \leq 1$$

$$\text{Cross correlation rating} \quad 0 \leq g_2 \leq 1$$

$$\text{Normalisation} \quad g_1 + g_2 = 1$$

4.2 Load case rating

The ratings of all signals of a load case C_{3_i} are combined to a load case rating C_{lc} . Every signal is assigned to an individual weighting factor (wf_i , **WF**). A load case rating is the weighted sum of its signal ratings.

Load case	$C_{lc} = \sum (wf_i \cdot C_{3_i})$
-----------	--------------------------------------

Normalisation	$\sum wf_i = 1$
---------------	-----------------

If a load case contains no signals but sub-load cases, then the load case rating is the weighted (wf_{slc_i} , **WF_SLC**) sum of the sub-load case ratings C_{slc_i} .

Load case	$C_{lc} = \sum (wf_{slc_i} \cdot C_{slc_i})$
-----------	--

Normalisation	$\sum wf_{slc_i} = 1$
---------------	-----------------------

4.3 Sub-load case rating

The ratings of all signals of a sub-load case C_{3_i} are combined to a sub-load case rating C_{slc} . Every signal is assigned to an individual weighting factor (wf_i **WF**). So a sub-load case rating is the weighted sum of its signal ratings.

Sub-load case	$C_{slc} = \sum (wf_i \cdot C_{3_i})$
---------------	---------------------------------------

Normalisation	$\sum wf_i = 1$
---------------	-----------------

4.4 Overall rating

The ratings of all load cases are combined to an overall rating. This rating score C is the weighted (wf_{lc_i} , **WF_LC**) sum of all load case ratings.

Overall rating	$C = \sum (wf_{lc_i} \cdot C_{lc_i})$
----------------	---------------------------------------

Normalisation	$\sum wf_{lc_i} = 1$
---------------	----------------------

5 Corridor method

The corridor metric assesses the compliance of a curve with corridors. One of the advantages of the corridor rating is the simplicity and the clearness of the algorithm. It reflects criteria which are used intuitively in engineering judgement. Figure 3 shows the curves used by the corridor method exemplary. Around the reference curve (black) an inner (green) and an outer (grey) corridor is drawn. The width of the corridors is mainly influenced by the parameter Y_{norm} .

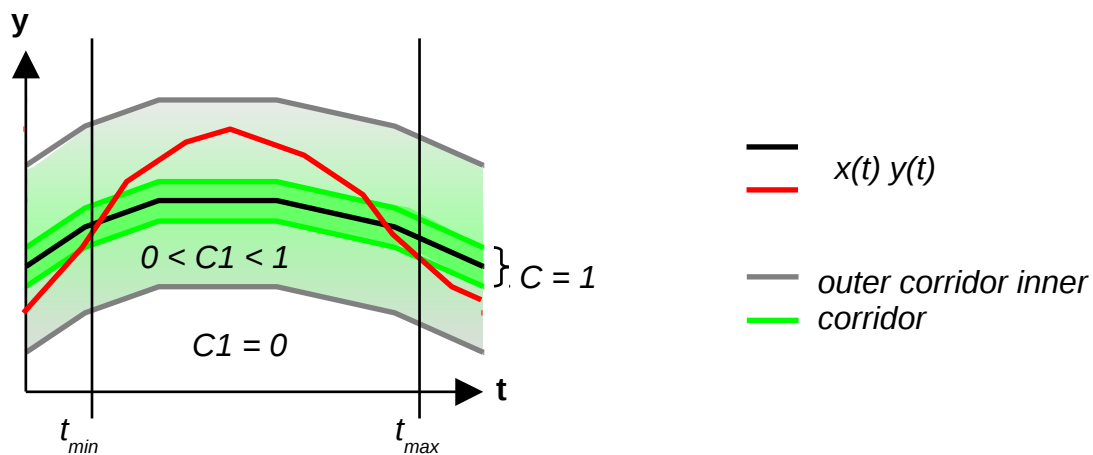


Figure 3: Corridor method

If the comparison (red) curve is within the inner corridor, the result is “1” (perfect match). If the comparison curve is outside the outer corridor, the result is “0” (poor correlation). In between the rating is interpolated.

5.1 Requirements

The reference curve and the comparison curve must both be defined between t_{min} and t_{max} . Both have n supporting points in this interval.

5.2 Determination of the corridors

The inner ($\delta_i(t)$) and out ($\delta_o(t)$) corridor can be calculated automatically or set by the user. The following requirement shall be kept to draw the corridors:

Basic requirement	$\delta_i(t) < \delta_o(t)$
Inner lower corridor	$\delta_{i_lower}(t) = x(t) - \delta_i(t)$
Inner upper corridor	$\delta_{i_upper}(t) = x(t) + \delta_i(t)$
Outer inner corridor	$\delta_{o_lower}(t) = x(t) - \delta_o(t)$
Outer lower corridor	$\delta_{o_upper}(t) = x(t) + \delta_o(t)$

5.2.1 Automatic corridors

As shown in Figure 3, two corridor areas are defined around the reference curve. The simplest method to define corridors of constant width is shown below:

Inner corridor	$\delta_i(t) = a_0 \cdot Y_{norm}$
Outer corridor	$\delta_o(t) = b_0 \cdot Y_{norm}$
Boundary conditions	$0 \leq a_0 \leq 1$, $0 \leq b_0 \leq 1$, and $a_0 < b_0$

a_0 and b_0 define the relative half width of the corridors and can be defined globally (section 9.1.4) and locally (section 9.2.3 and 9.3.3) in the control file. The parameter Y_{norm} is the reference amplitude that is used to calculate the corridors. It is either a user-defined value or the maximum absolute amplitude of the reference curve (setting “extremum”).

$$Y_{norm} \text{ set to “extremum”} \quad Y_{norm} = \max(\min(x(t)), \max(x(t)))$$

If the reference curve is the average curve of several tests (n_{Test}), it is possible to use the standard deviation $\sigma(t)$ to calculate the corridor width. The standard deviation, influenced by the scaling parameters a_s and b_s , is added to the corridors. A corridor can also be defined by using the standard deviation only (**Y_NORM** must be set to “0”). However, if $\sigma(t)$ is used to influence the corridors, then the width of the corridors is not constant.

Standard deviation	$\sigma(t) = \sqrt{\frac{\sum_{i=1}^{n_{Test}} (x_i(t) - x(t))^2}{n_{Test}}}$
--------------------	---

Inner corridor	$\delta_i(t) = a_0 \cdot Y_{norm} + a_s \cdot \sigma(t)$
Outer corridor	$\delta_o(t) = b_0 \cdot Y_{norm} + b_s \cdot \sigma(t)$
Boundary conditions	$0 \leq a_0 \leq 1$, $0 \leq b_0 \leq 1$, and $a_0 < b_0$

If a test curve has values outside the outer corridor, the corridor is locally enlarged to include all test curves.

5.2.2 User-defined corridors

CORApplus offers the possibility to work with user-defined corridors, provided in a separate file. They must be specified for the complete interval of evaluation and it is essential that the condition $\delta_i(t) < \delta_o(t)$ is maintained for the complete interval. Furthermore, the curves must be use the output unit system.

5.3 Rating

At first the corridor rating is calculated at every time step t_i within the interval of evaluation $[t_{min}, t_{max}]$. The total corridor rating C_1 is the average of all time step ratings c_i .

$$c_i = \begin{cases} 1 & \text{if } |y(t_i) - x(t_i)| < \delta_i(t) \\ \left(\frac{\delta_o(t) - |y(t_i) - x(t_i)|}{\delta_o(t) - \delta_i(t)} \right)^k & \text{with } k \in N_{>0} \\ 0 & \text{if } |y(t_i) - x(t_i)| > \delta_o(t) \end{cases}$$

$$C_1 = \frac{\sum_{i=1}^n c_i}{n} \quad \text{with} \quad 0 \leq C_1 \leq 1$$

The interpretation of the equation of c_i is simple. If the distance between reference and comparison curve is bigger than δ_o , the correlation is poor ($c_i = 0$). If the distance between reference and comparison curve is smaller than δ_i , the correlation is good ($c_i = 1$). In between an interpolation ("0" to "1") is performed. The exponent k describes decline of the

rating between “1” and “0”. The bigger k , the higher is the decline of the rating of a curve compared to its distance to the inner corridor.

5.4 Disadvantages

The corridor method compares the values of the two curves at each time step. A distortion of the phase can lead to a very poor rating.

6 Cross correlation method

The cross correlation method analyses three characteristics of a signal – progression V (Figure 4), phase shift P (Figure 5), and size G (Figure 6). The result of each sub rating ranges from “0” (no correlation) to “1” (perfect match). The cross correlation rating C_2 is the sum of those individually weighted sub-ratings.

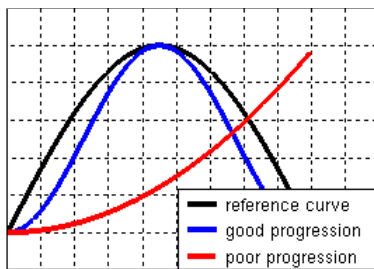


Figure 4: Progression rating

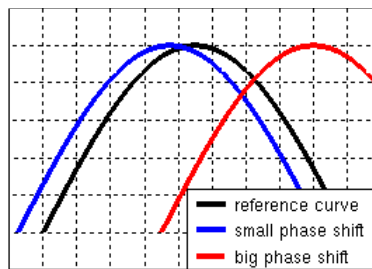


Figure 5: Phase shift rating

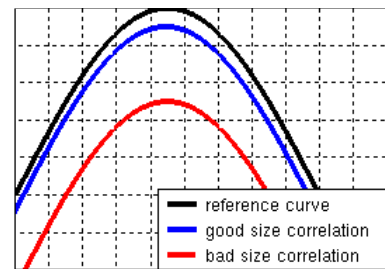


Figure 6: Size rating

6.1 Requirements

The reference curve and the comparison curve shall be both defined between t_{min} and t_{max} . Both have n supporting points of a constant Δt within this interval.

6.2 Procedure

As mentioned above, the corridor method can lead to poor results if the two curves have a time shift. Therefore, this metric tries to eliminate this problem first. So the cross correlation metric starts with moving the reference curve by multiples of Δt (Figure 7). For each of these moved states the cross correlation value K is calculated. K ranges from “-1” to “1” and is a rating of the progression of the curve. The time shift with the maximum K value is used to calculate the three parts of the cross correlation rating.

$$\text{Maximum cross correlation } K = \max(K_{\delta_i}) \text{ with } \delta = m_0 \cdot \Delta t$$

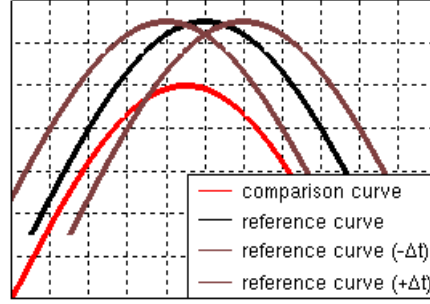


Figure 7: Phase shift to calculate the cross correlation

6.3 Maximum cross correlation

The cross correlation K_{xy} is calculated according to the following equation:

$$K_{xy}(m) = \frac{\sum_{i=0}^{n-1} x(t_{min} + (m+i) \cdot \Delta t) \cdot y(t_{min} + i \cdot \Delta t)}{\sqrt{\left(\sum_{i=0}^{n-1} x^2(t_{min} + (m+i) \cdot \Delta t) \cdot \sum_{i=0}^{n-1} y^2(t_{min} + i \cdot \Delta t) \right)}} \quad \text{with} \quad -1 \leq K_{xy} \leq 1$$

The reference curve is shifted by an alterable time shift of $m \cdot \Delta t$ with $m=0, 1, -1, 2, \dots$. The range of m and thus, the maximum time shift of the reference curve is limited by the global parameter INT_{min} . Minimum and maximum time shift are calculated by the following equations:

$$t_{shiftmin} = -(1 - INT_{min}) \cdot (t_{max} - t_{min})$$

$$t_{shiftmax} = (1 - INT_{min}) \cdot (t_{max} - t_{min})$$

Therefore, the signals $x(t)$ and $y(t)$ are at least overlapping within the interval $t_{overlap}$.

$$t_{overlap} \geq INT_{min} \cdot (t_{max} - t_{min}) \quad \text{with} \quad (0 < INT_{min} < 1)$$

The range of m is obtained as follows:

$$m_{min} = \frac{t_{shiftmin}}{\Delta t}$$

$$m_{max} = \frac{t_{shiftmax}}{\Delta t}$$

The parameter n to calculate $K_{xy}(m)$ is not constant, but is reduced to $n_{min} = \text{INT}_{min} \cdot n$ for m_{min} and m_{max} .

The examples shown in section 6.8 may help to understand the time shift of the reference curve. The cross correlation K is the maximum of all $K_{xy}(m)$:

$$K = K_{xy}(m_0) = \max_m (K_{xy}(m))$$

6.4 Progression rating V

The progression rating is derived from K :

$$\text{Progression rating} \quad V = \left(\frac{1}{2} (K + 1) \right)^{k_v} \quad \text{with } k_v \in N_{>0} \text{ and } 0 \leq V \leq 1$$

The exponent k_v describes the decline of the rating between “1” and “0”. A linear, quadratic or any other regression relationship can be defined accordingly.

6.5 Phase shift rating P

The phase shift rating is controlled by the parameters D_{MIN} and D_{MAX} . Both are used to calculate the signal specific time values δ_{min} and δ_{max} .

$$\delta_{min} = D_{MIN} \cdot (t_{max} - t_{min}) \quad \text{with } 0 < D_{MIN} \leq 1$$

$$\delta_{max} = D_{MAX} \cdot (t_{max} - t_{min}) \quad \text{with } 0 < D_{MAX} \leq 1$$

D_{min} and D_{max} can be specified globally or locally. The phase shift rating P is calculated at the maximum cross correlation K and the corresponding time shift δ as follows:

$$\text{Phase shift rating} \quad P = \begin{cases} 1 & \text{if } |\delta| < \delta_{min} \\ \left(\frac{|\delta_{max} - |\delta||}{\delta_{max} - \delta_{min}} \right)^{k_p} & \\ 0 & \text{if } |\delta| > \delta_{max} \end{cases} \quad \text{with } k_G \in N_{>0}$$

It is similar to the rating done for the corridor method with two given boundary values. The exponent k_p describes the decline of the rating between “1” and “0”.

6.6 Size rating G

The size rating is done by comparing the square of the areas between the curves and the time axis. Since both curves have equidistant supporting points, the following ratio can be calculated:

$$F_x[t_{min}, t_{max}]$$

$$F_y[t_{min} + \delta, t_{max} + \delta]$$

$$\frac{F_x}{F_y} = \frac{\sum_{i=1}^n x^2(t_{min} + \delta + i \cdot \Delta t)}{\sum_{i=1}^n y^2(t_{min} + i \cdot \Delta t)}$$

The following equation is used to calculate the size rating:

$$\text{Size rating} \quad G = \begin{cases} \left(\frac{F_x}{F_y} \right)^{k_G} & \text{if } F_y > F_x \\ \left(\frac{F_y}{F_x} \right)^{k_G} & \end{cases} \quad \text{with } k_G \in N_{>0}$$

The exponent k_G describes the decline of the rating between “1” and “0”.

6.7 Rating

The cross correlation rating C_2 is a combination of the three sub-ratings V , P , and G .

$$\text{Cross correlation rating } C_2 = g_V \cdot V + g_P \cdot P + g_G \cdot G$$

$$\text{Boundary condition } 0 \leq (g_V, g_P, g_G) \leq 1$$

The weighting factors g_V , g_P and g_G can be defined globally (G_V , G_P , G_G) and locally (g_V , g_P , g_G). To guarantee $0 \leq C_2 \leq 1$, the condition $g_V + g_P + g_G = 1$ must be met.

6.8 Example of the maximum time shift calculation

Figure 8 and Figure 9 depict an example of the time shift calculation. The baseline condition is shown on the left and the shifted curves with i_{min} and i_{max} on the right.

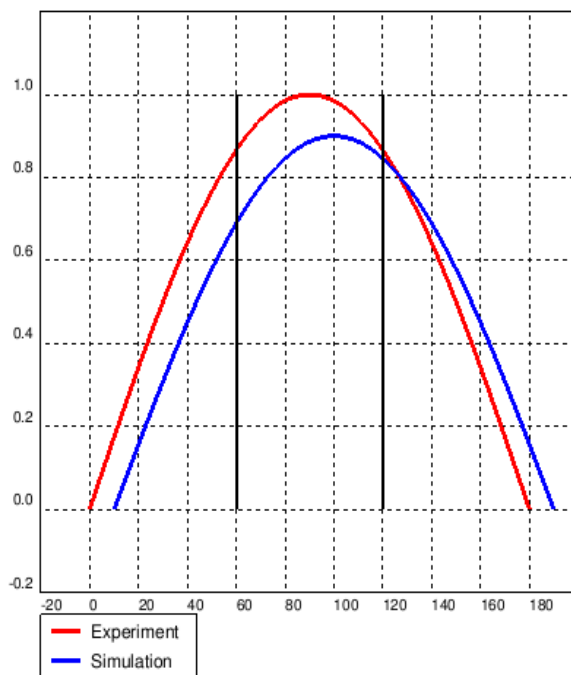


Figure 8: Baseline condition

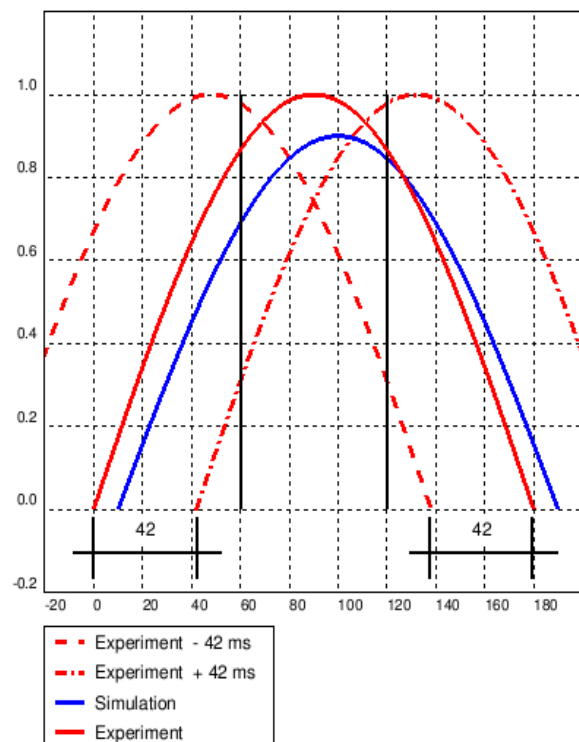


Figure 9: Shifted test curve

The following parameters are defined:

T_MIN/T_MAX 60 120
INT_MIN 0.3

According to the equations in shown in section 6.3, the minimum overlap is calculated as follows:

$$t_{shiftmin} = -(1 - INT_{min}) \cdot (t_{max} - t_{min}) = -(1 - 0.3) \cdot (120 - 60) = -42$$

$$t_{shiftmax} = (1 - INT_{min}) \cdot (t_{max} - t_{min}) = (1 - 0.3) \cdot (120 - 60) = 42$$

The time shift reduced the overlap of reference curve and interval of evaluation to $t_{overlap}$:

$$t_{overlap} = INT_{min} \cdot (t_{max} - t_{min}) = 0.3 \cdot (120 - 60) = 18$$

6.9 Disadvantages

It is essential to select a reasonable size of the evaluation interval and the maximum phase shift. Otherwise small parts of the curves which have a good similarity but large phase-shift can influence the cross correlation values significantly. Especially periodic functions could be affected.

7 Biofidelity rating according to ISO/TR 9790

The keyword **METHOD** must be set to "ISO9790" to enable the biofidelity rating according to ISO/TR 9790 [2]. Furthermore, the required corridor files must be specified in the signal parameter block (parameter *a_t*). The signal parameter *Y_norm* selects the rating metric. The allowed terms of *Y_norm* are described in the following sections.

The complete set of tests defined in ISO/TR 9790 can be applied by defining a load case for each body region and a sub-load case for each test. The weighting factors and corridors are not hard-coded in the CORApplus software. They have to be specified in the control file. The definition of those corridor files is described in section 10.

7.1 Corridor method

Two corridor curves (upper and lower) define the inner corridor. The maximum distance between the two corridor curves is added to the upper curve and subtracted from the lower curve to get the outer corridor. If the test curve values are totally within the inner corridor, the rating is "10". A rating of "5" is given, if all curve values are within the limits of the outer corridor. Otherwise the rating is "0". Figure 10 shows the ISO-corridor rating exemplary.

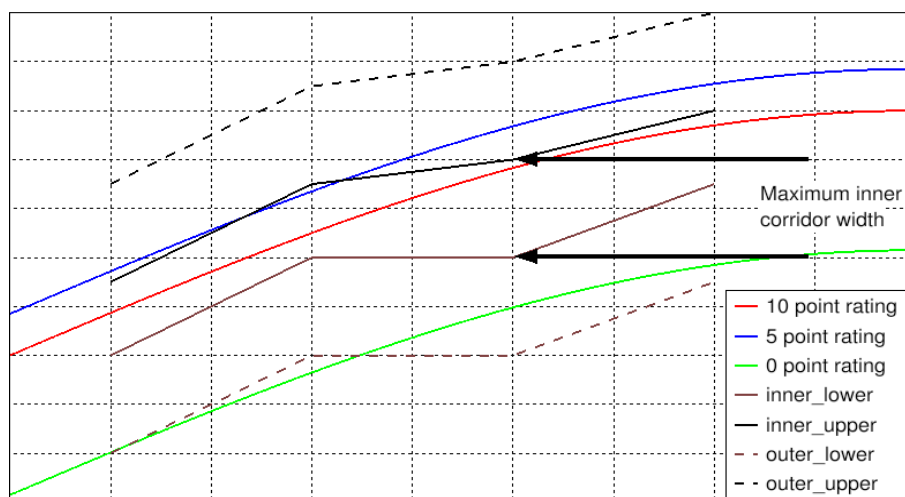


Figure 10: Example of the corridor rating

The term "corridor" selects the corridor rating procedure of ISO/TR 9790.

7.2 Peak method

This method requires corridors at a certain level, shown in Figure 11. The rating is calculated like the corridor rating, but only the peak value (maximum or minimum) of a response is evaluated.

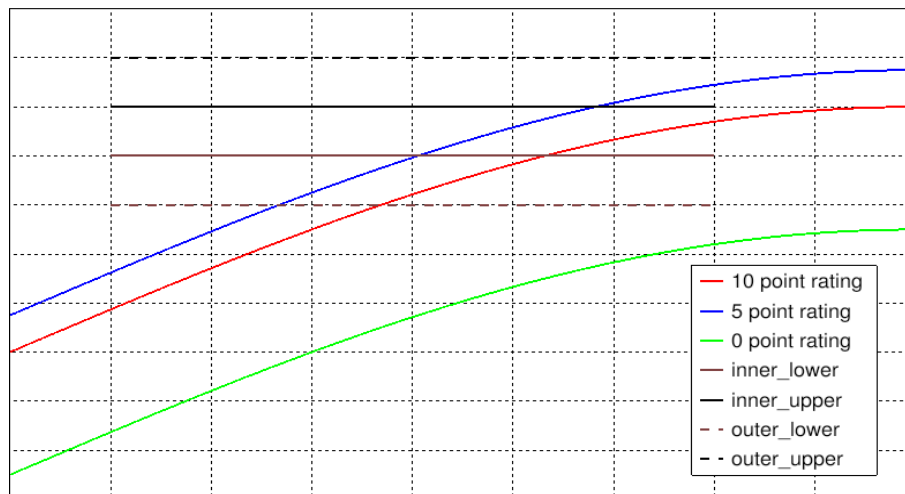


Figure 11: Example of the peak rating

The term “peak” enables the peak rating procedure.

7.3 High/low peak method

This method evaluates only the peak value of a curve and not the compliance with a corridor. If the peak value is above (“peak_low”) or below (“peak_high”) a specified threshold, the rating is set to “10”. Otherwise the rating is “0”. An example of this method is shown in Figure 11.

This method is enabled by the terms “peak_high” or “peak_low”.

7.4 Alternative high/low peak method

The only difference to the method described in section 7.3 is the possibility to get an additional rating of “5”. This result is given, if half or twice of a threshold is reached. Figure 12 shows an example of this method.

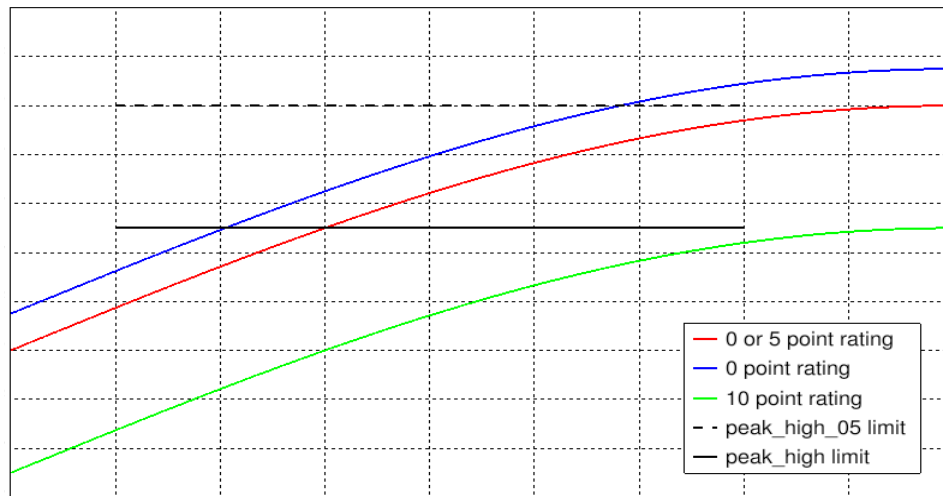


Figure 12: Example of the peak high and the alternative peak high rating

The terms “peak_high_5” or “peak_low_5” enables an alternative peak rating algorithm.

8 ISO18571 rating

The standard only defines the rating of two signals. How to filter or which part of a signal is to be rated is not specified in the ISO18571. The 'normal' Cora procedure is used for these curve preparations as well as for calculating a mean TEST curve. The (sub)loadcase and total rating is not part of the ISO18571 but shown anyway, but the values are put in brackets.

The ISO rating uses the same syntax in the ASCII control files, but ignores most of the rating parameters which are implemented as defined by the ISO.

The ISO recommends to have a signal sampling frequency of 10kHz. This is not checked by CORApplus.

To use the ISO18571 rating add the Keyword **METHOD ISO18571** to a loadcase or subloadcase section of the CORApplus control file.

8.1 Symbols and subscripts

The ISO uses some different symbols.

$x, x(t)$ the reference curve equals $T, T(t)$ the TEST curve

$y, y(t)$ the reference curve equals $C, C(t)$ the CAE curve

Y_{norm} the curve reference value equals T_{norm}

t_{min}, t_{max} the evaluation interval equals t_{start}, t_{end}

8.2 The Corridor Rating

The calculation of the corridor rating is the same as the Cora Corridor rating. The ISO18571 uses fix values for some parameters and sometimes other identifiers:

- T_{norm} and is not to be specified, but always calculated

$$T_{norm} = \max\{ \min(T) , \max(T) \}$$

- a_0 and b_0 are set to 0.05 and 0.5

- The corridors are calculated using T_{norm} and a_0 and b_0 . The usage of corridor files is not part of the standard.
- The exponent K is called K_z and has the value of 2.

8.3 The cross-correlation Rating

The cross-correlation rating consist of the phase, magnitude, and slope scores.

At the beginning the phase score is calculated. As in Cora method the CAE curve is phase-shifted by multiple of the sampling interval Δt .

The maximum allowed time-shift is $\epsilon_{pmax} \cdot (t_{max} - t_{min})$ with $\epsilon_{pmax} = 0.2$

With N the number of sampling intervalls the maximum number of time-shifts is

$$n = \epsilon_{pmax} \cdot N$$

The correlation for each time-shift is then calculated according to the formula

$$P_L(m) = \frac{\sum_{i=0}^{n-1} [(C(t_{start} + (m+i) \cdot \Delta t) - \bar{C}(t)) \cdot (T(t_{start} + i \cdot \Delta t) - \bar{T}(t)))]}{\sum_{i=0}^{n-1} [C(t_{start} + (m+i) \cdot \Delta t) - \bar{C}(t)]^2 \cdot \sum_{i=0}^{n-1} [T(t_{start} + i \cdot \Delta t) - \bar{T}(t)]^2}$$

when shifted to the left and

$$P_R(m) = \frac{\sum_{i=0}^{n-1} [(C(t_{start} + i \cdot \Delta t) - \bar{C}(t)) \cdot (T(t_{start} + (m+i) \cdot \Delta t) - \bar{T}(t)))]}{\sum_{i=0}^{n-1} [C(t_{start} + i \cdot \Delta t) - \bar{C}(t)]^2 \cdot \sum_{i=0}^{n-1} [T(t_{start} + (m+i) \cdot \Delta t) - \bar{T}(t)]^2}$$

when shifted to the right. The time-shift n_ϵ where the cross-correlation value

$P = \max(P_L, P_R)$ reaches its maximum value is used to calculate the phase shift score

$$E_P = \begin{cases} 1 & \text{if } n_\epsilon = 0 \\ \left(\frac{\epsilon_{pmax} \cdot N - n_\epsilon}{\epsilon_{pmax} \cdot N} \right)^{k_p} & \text{with } k_p = 1 \\ 0 & \text{if } n_\epsilon > \epsilon_{pmax} \cdot N \end{cases}$$

The magnitude score is calculated on the remaining overlapping parts of the curves after the time-shift due to n_ϵ is made. The ISO18571 calls the remaining part of the CAE curve truncated and shifted CAE curve C^{ts} . The remaining part of the TEST curve is the truncated TEST curve T^t .

With T^t and C^{ts} a dynamic time warping (DTW) is performed. Meinard Müller 74 calls the used algorithm described in the ISO18571 'Classical DTW'.

For the DTW a cost matrix d is calculated which is a $n \times n$ matrix with n the number of remaining supporting points.

$$d(i, j) = (C^{ts}(i) - T^t(j))^2$$

This matrix can be transformed into the cumulative cost matrix dtw

$$dtw = \begin{cases} d(i, j) & i=1, j=1 \\ d(i, j) + dtw(i, j-1) & i=1 \\ d(i, j) + dtw(i-1, j) & j=1 \\ d(i, j) + \min(dtw(i-1, j-1), dtw(i-1, j), dtw(i, j-1)) & otherwise \end{cases}$$

The total cost matrix dtw is used to calculate the optimal warping path. This path starts at $dtw(n, n)$ and moves to $dtw(1, 1)$ according to the formula:

$$[i^w - 1, j^w - 1] = \begin{cases} [i-1, 1] & j=1 \\ [1, j-1] & i=1 \\ [i-1, j] & dtw(i-1, j) = \min(dtw(i-1, j-1), dtw(i-1, j), dtw(i, j-1)) \\ [i, j-1] & dtw(i, j-1) = \min(dtw(i-1, j-1), dtw(i-1, j), dtw(i, j-1)) \\ [i-1, j-1] & dtw(i-1, j-1) = \min(dtw(i-1, j-1), dtw(i-1, j), dtw(i, j-1)) \end{cases}$$

With the indices of the warping path the warped curves T^{t+w} and C^{ts+w} can be created.

The result is a (much) longer signal ($n \leq n_w \leq 2*n-1$) which has a much supporting points at the warping path. The x values (time) of the warped curves are not of interest. With these two curves the magnitude error is defined as

$$\epsilon_{mag} = \frac{\|C^{ts+w} - T^{t+w}\|_1}{\|T^{t+w}\|_1}$$

The magnitude score is defined as

$$E_M = \begin{cases} 1 & \text{if } \epsilon_{mag} = 0 \\ \left(\frac{\epsilon_m - \epsilon_{mag}}{\epsilon_m} \right)^{k_M} & \\ 0 & \text{if } \epsilon_{mag} > \epsilon_m \end{cases} \quad \text{with } k_M = 1, \epsilon_m = 0.5$$

The slope score is based on the curves T^t and C^{ts} . Central difference are first used to calculate the slope, except the first point taking a forward difference and the

last point taking a backward difference.

The slope curves are averaged by a 9 point moving average. At both ends of the curve the number of points is reduced to 1, 3, 5, or 7 points, so that always the equal number of points left and right of the current point are taken for the average value. The averaged slopes form the new curves T^{t+d} and C^{ts+d} which are used to calculate the slope error.

$$\epsilon_{slope} = \frac{\| (C^{ts+d} - T^{t+d}) \|_1}{\| (T^{t+d}) \|_1}$$

The slope error is used to calculate the slope score.

$$E_S = \begin{cases} 1 & \text{if } \epsilon_{slope} = 0 \\ \left(\frac{\epsilon_s - \epsilon_{slope}}{\epsilon_s} \right)^{k_S} & \\ 0 & \text{if } \epsilon_{slope} > \epsilon_s \end{cases} \quad \text{with } k_S = 1, \epsilon_s = 2.0$$

8.4 Calculation of the overall ISO rating

The total ISO rating is a combination of the four single ratings.

$$R = W_Z * Z + W_P * E_P + W_M * E_M + W_S * E_S$$

The weighting factors are set to

$$W_Z = 0.4, W_P = 0.2, W_M = 0.2 \wedge W_S = 0.2$$

The result R is categorized in four ranges

Rank	Grade	Rating R
1	Excellent	$R > 0.94$
2	Good	$0.80 < R \leq 0.80$
3	Fair	$0.58 < R \leq 0.80$
4	Poor	$R \leq 0.58$

9 Control file

The ASCII control file uses a block format. Comment lines starts with a hashmark (“#”) and can be placed anywhere in the file. In-line comments can be placed at the end of a command and shall start with a semicolon (“;”). It is not possible to split commands to several lines. The maximum allowed line length is 4095 characters.

The keywords and parameters must be separated by at least two blank characters. Blanks at the beginning or at the end of a line will be cut automatically.

The file structure is:

- global parameters

- load case 1

 - parameters of load case 1

 - curve files for load case 1

 - signals of load case 1

- load case n

 - parameters of load case n

 - sub-load case n.1

 - parameters of sub-load case n.1

 - curve files for sub-load case n.1

 - signals of sub-load case n.1

 - sub-load case n.m

 - parameters of sub-load case n.m

 - curve files for sub-load case n.m

 - signals of sub-load case n.m

At least one load case must be defined. The use of sub-load cases is optional.

9.1 Global parameters

Global parameters define or adjust settings which are used by all load cases and sub-load cases. They can be overruled by parameters of load cases and sub-load cases. Those parameters are also called local parameters. Many parameters of CORApplus can be defined globally and locally. If local parameter is set to “x”, the global setting is used.

The global parameters block starts with the keyword **BEGIN GLOBAL_PARAMETERS** and is closed with the keyword **END GLOBAL_PARAMETERS**. The order of the parameters within this block is arbitrary.

9.1.1 Parameters to calculate the interval of evaluation

A_EVAL <real> Domain: $[0,1]$

Expand the automatically calculated interval of evaluation (see α_{eval} in section 3.2)

A_THRES <real> Domain: $[0,1]$

This parameter is used in conjunction with the keyword **Y_NORM** to calculate the starting time of the interval of evaluation automatically (see a_{thres} in section 3.2).

B_THRES <real> Domain: $[0,1]$

This keyword is used in conjunction with **Y_NORM** to calculate the ending time of the interval of evaluation automatically (see b_{thres} in section 3.2).

B_DELTA_END <real> Domain: $[0,1]$

This keyword is used to shorten the end of the interval of evaluation if it is calculated automatically (see b_{delta_end} in section 3.2).

Disable option: “0”

T_MIN/T_MAX <real> <real> Domain: $\{(-\infty, \infty), \text{automatic}\}$

This keyword defines the starting and the ending time of the interval of evaluation $[t_{min}, t_{max}]$ for all signals.

If the term “automatic” is used (for both or only one), CORApplus calculates reasonable intervals signal-wise (section 3.2). The parameters ***T_MIN/T_MAX*** use the system of unit specified in the file `output_units.txt`.

Y_NORM <real> Domain: $\{(-\infty, \infty), \text{extremum}\}$

The keyword defines the reference value for the calculation of the corridor widths for all signals. The term “extremum” enables the use of the maximum absolute amplitude of the reference curve (see Y_{norm} in section 5.2.1).

ISONAME_1-2/11-12 <string> <string> Domain: {YES, NO}

A signal name according to ISO/TS 13499 codes several information about a signal. So the 1st character specifies the test object, the 2nd character the seating position and the 11th and 12th character specify additional information about the location of the measurement (e.g. type of the dummy used).

CORApplus offers the possibility to ignore (set to “NO”) those characters when analysing and compiling the signal names of the reference and comparison data.

Example:

ISONAME_1-2/11-12 YES NO

S1HEAD0000***H3***ACXP is equivalent to ***S1***HEAD0000***H***FACXP

S1HEAD0000***H3***ACXP is not equivalent to ***S3***HEAD0000***H***FACXP

9.1.2 Parameters to format the HTML report

DES_GLO <string>

Description of the evaluation (title)

DES_MOD <string>

Additional description of the evaluation (sub-title)

FONT_SMALL <integer> Domain: $N_{>0}$

Pixel size of the small fonts used in the signal pages of the HTML report

FONT_LARGE <integer> Domain: $N_{>0}$

Pixel size of the large fonts used in the signal pages of the HTML report

OUTPUT_FORMAT <character> Domain: {H, L, P}

This keyword defines the format of the exported ASCII curves. It can be set to Hyper-Graph, LS-PrePost or PAM-VIEW. Only the first character is recognised – “H”, “L”, or “P”.

PreT_LC/PostT_LC <real> <real> Domain: { $R_{>0}$, -1, -2}

This keyword is used to define the shown domain of the x-axis in the charts. The interval of evaluation is the baseline domain but it can be expanded by a percentage of the length of this interval:

$$\text{Start} \quad t_s = t_{min} - PreT_{LC}(t_{max} - t_{min})$$

$$\text{End} \quad t_e = t_{max} - PostT_{LC}(t_{max} - t_{min})$$

Some specific values enables some pre-defined settings. If the first value is set to “-1”, the x-axis starts at the first time sample of the reference signal. When using “-2”, the x-axis starts at “0”. If the second value is set to “-1” the x-axis ends at the last time sample of the reference signal.

WF_NORM <string> Domain: {YES, NO}

Switch to enable or disable the output of normalised weighting factors.

9.1.3 Small signals

MIN_NORM <real> Domain: [0,1]

This keyword controls the mechanism to detect major and minor axes of a sensor with multiple directions in space (e.g. x, y, and z acceleration of a dummy head). Minor axes can be handled by special procedures to reduce their influence on a load case or sub-load case rating.

If signal names specified in a load case or a sub-load case differ only in its direction in space (x, y, and z), then those signals are treated as a sensor with multiple axes and the **MIN_NORM** algorithm can be applied.

Disable option: “0”

CORApplus searches the reference data set or if more than one reference data set is given, the average reference data set, for signals that differ only in direction in space

(sensor with multiple axes). If so, the absolute maximum amplitude of every signal of such a sensor is calculated ($MaxX$, $MaxY$, and $MaxZ$). The maximum of those values is $MaxT$ and is treated as the major axis of this sensor.

$$MaxT = \max(MaxX, MaxY, MaxZ)$$

An axis of a sensor is treated as a minor axis signal if the following condition is met:

$$(MaxX, MaxY, MaxZ) < (MIN_{NORM} \cdot MaxT)$$

The following changes are applied to minor axis signals:

- Interval of evaluation: interval of $MaxT$ is used
- Corridor method: $Y_{norm} = MIN_{NORM} \cdot MaxT$
- Cross correlation method: is calculated but its not considered in the signal evaluation (weighting factor g_2 is set to "0")

9.1.4 Parameters to tune the corridor method

a_0/b_0 <real> <real> Domain: $[0,1]$

Relative width of the inner (**a_0**) and outer (**b_0**) corridor (see a_0 and b_0 in section 5.2)

a_sigma/b_sigma <real> <real> Domain: $[0,1]$

This keyword defines two factors to include the standard deviation $\sigma(t)$ of the reference signals in the calculation of the inner (**a_sigma**) and the outer (**b_sigma**) corridor (see a_s and b_s in section 5.2). This function only makes sense if multiple reference curves are given.

Disable option: "0"

G_1 <real> Domain: $R_{>0}$

Weighting factor of the corridor rating C_1 (see g_1 in section 4)

K <integer>Domain: $N_{>0}$

Exponent factor (progression) for the calculation of the corridor rating (see k in section 5.3)

9.1.5 Parameters to tune the cross correlation method

D_MIN <real>Domain: $[0,1]$

The parameter limits the allowed phase shift to calculate the phase shift rating (see D_{MIN} in section 6.5). **D_MIN** is only applied to the interval of evaluation.

D_MAX <real>Domain: $[0,1]$

The parameter limits the allowed phase shift to calculate the phase shift rating (see D_{MAX} in section 6.5). **D_MAX** is only applied to the interval of evaluation.

G_V <real>Domain: $R_{>0}$

Weighting factor of the progression rating V (see g_V in section 6.7)

G_P <real>Domain: $R_{>0}$

Weighting factor of the phase shift rating P (see g_P in section 6.7)

G_G <real>Domain: $R_{>0}$

Weighting factor of the size rating G (see g_G in section 6.7)

G_2 <real>Domain: $R_{>0}$

Weighting factor of the cross correlation rating C_2 (see g_2 in section 4)

INT_MIN <real>Domain: $[0,1]$

This parameter defines the maximum time shift used by the cross correlation method. The result of $INT_{min} \cdot (t_{max} - t_{min})$ is the minimum remaining overlapping time of reference curve and comparison curve (see INT_{min} in section 6.3).

K_V <integer>Domain: $N_{>0}$

Exponent factor (progression) of the calculation of the progression rating V (see k_V in section 6.4)

K_P <integer>Domain: $N_{>0}$

Exponent factor (progression) of the calculation of the phase shift rating P (see k_p in section 6.5)

K_G <integer>Domain: $N_{>0}$

Exponent factor (progression) of the calculation of the size rating G (see k_G in section 6.6)

9.1.6 Definition of the weighting factors by physical dimensions

DIM_WEIGHTS <string> <real> <real> <real> <real> <real>

The 13th and 14th character of an ISO signal name define the physical dimension of a signal. For example “AC” stands for linear accelerations. The weighting factors ***G_V***, ***G_P***, ***G_G***, ***G_1*** and ***G_2*** can be pre-defined for all signals of the same physical dimension.

The following parameters can be defined (sequentially in the following order):

- physical dimension, terms according to ISO/TS 13499
- G_V , section 9.1.5
- G_G , section 9.1.5
- G_P , section 9.1.5
- G_1 , section 9.1.4
- G_2 , section 9.1.5

This optional keyword can be used only once for each physical dimension. It overrules all other global and local settings.

9.1.7 Example of a global parameter block

The parameters of the corridor method are printed in blue and parameters of the cross correlation method are printed in red.

```
BEGIN GLOBAL_PARAMETERS
# comment
A_EVAL           0.00 ; another comment
A_THRES          0.05
```

```

B_THRES          0.07
B_DELTA_END      0.1
T_MIN/T_MAX      automatic    1.0
Y_NORM           extremum
DES_GLO          what it is
DES_MOD          Model 42
FONT_SMALL       13
FONT_LARGE       16
ISONAME_1-2/11-12 YES    NO
OUTPUT_FORMAT    LSPOST
PreT_LC/PostT_LC 0.0    0.1
WF_NORM          YES
MIN_NORM         0.20
a_0/b_0          0.7    0.9
a_sigma/b_sigma  0.2    0.5
K                2
G_1              0.50
D_MIN            0.05
D_MAX            0.10
INT_MIN          0.3
K_V              1.0
K_G              1.0
K_P              1.0
G_V              0.33
G_G              0.33
G_P              0.33
G_2              0.50
DIM_WEIGHTS      AC    0.4    0.4    0.2    0.55    0.45
END GLOBAL_PARAMETERS

```

9.2 Load case parameters

The definition of a load case starts with **BEGIN LOADCASE** and ends with **END LOADCASE**. Within this block a curve file block (between **BEGIN DATAFILES** and **END DATAFILES**) and a signal block (between **BEGIN SIGNALS** and **END SIGNALS**) shall be defined.

9.2.1 General Parameters

NAM_LC <string>

Name of the load case (title)

DES_LC <string>

Additional description of the load case (sub-title)

METHOD <string> Domain: {CORA, ISO9790, ISO18571}

This keyword selects the type of rating performed for all signals of a load case. It can be either set to “ISO18571” (32), “ISO9790” (29) or to “CORA”. The later (default) uses the corridor metric (section 5) and cross-correlation metric (section 6).

This keyword is optional.

MinOrd_LC <real> Domain: $R_{>0}$

This factor scales the minimum value of the y-axis of the charts shown in the HTML report. The default value is “1.0”.

$$y_s = MinOrd_{lc} \cdot y_{min}$$

This keyword is optional.

MaxOrd_LC <real> Domain: $R_{>0}$

This factor scales the maximum value of the y-axis of the charts shown in the HTML report. The default value is “1.0”.

$$y_s = MaxOrd_{lc} \cdot y_{max}$$

This keyword is optional.

PreT_LC <real> Domain: { $R_{>0}$, x }

See also the description of the global parameter **PreT_LC/PostT_LC** in section 9.1.2. If the term “x” is specified, then the global setting is used.

This keyword is optional.

PostT_LC <real> Domain: { $R_{>0}$, x }

See also the description of the global parameter **PreT_LC/PostT_LC** in section 9.1.2. If the term “x” is specified, then the global setting is used.

This keyword is optional.

WF_LC <real> Domain: $R_{>0}$

Weighting factor of the load case (section 4)

9.2.2 Data file block parameters

A data file is a file that contains at least one curve. The supported file formats are ISO/TS 13499 (ISO-MME), Diadem, HyperGraph ASCII, LS-PrePost Curve and PAM-VIEW ASCII Format 3 (section 2.1). Each data file must be specified in a single line with some additional parameters. File names may not contain blanks. The shown order of the parameters must be kept. **The order is changed since Cora 4.0!**

<Name of the data file> Type: string
Name of the data file

<Time shift> Type: real, Domain: $(-\infty, \infty)$
The time shift is applied to all curves of a data file. It must be specified in the unit of the data file.

<System of units> Type: string
The physical unit of length, mass, and time have to be specified. They must be separated by "-" or ".". If displacements (ISO code "DS") use a different length unit than distances (ISO code "DC"), a second unit of the length can be specified in front of the other units.

<Dimension of linear accelerations> Type: string, Domain: {YES, NO}
CORApplus accepts [g] as unit of linear accelerations ($1 \text{ g} = 9.81 \text{ m/s}^2$). If the accelerations of the data file are using [g], the term "YES" must be specified otherwise "NO" is required.

Example:

#	Name	time-shift	unit	g
	FileTest1	0.0	mm.K.s	YES

9.2.3 Signal block parameters

All signal parameters must be specified sequentially in a single line and the following order.

<Name> Type: string
Name of the signal. Signal names, that comply with ISO/TS 13499 should be used.

<*WF*> Type: real, Domain: $R_{>0}$

Signal weighting factor (see *wf* in section 4)

If **METHOD** CORA:

<*Y_norm*> Type: real, Domain: $\{(-\infty, \infty), \text{extremum}\}$

This parameter is used to calculate the interval of evaluation as well as the corridors of the corridor method (see Y_{norm} in section 5.2.1 and 9.1.1). *Y_norm* can be a number or set to “extremum” (section 9.1.1).

If **METHOD** ISO9790:

<*Y_norm*> Type: string

Specifies the rating method (section 7)

<*t_min*> Type: real, Domain: $\{(-\infty, \infty), x\}$

The parameter defines the start of the interval of evaluation. If a number is specified, the time unit is defined in the file `output_units.txt`. Besides a number, several expressions to enable special functions can be specified:

“x” the global setting is used (*T_MIN/T_MAX*, section 9.1.1)

“y” the setting of the first signal of the same (sub-)load case is used

“automatic” *t_min* is calculated automatically (section 3.2)

<*t_max*> Type: real, Domain: $\{(-\infty, \infty), x, y\}$

The parameter defines the end of the interval of evaluation. If a number is specified, the time unit is defined in the file `output_units.txt`. Besides a number, several expressions to enable special functions can be specified:

“x” the global setting is used (*T_MIN/T_MAX*, section 9.1)

“y” the setting of the first signal of the same (sub-)load case is used

“automatic” *t_max* is calculated automatically (section 3.2)

If **METHOD** CORA:

<*g_V*> Type: real, Domain: $\{R_{>0}, x\}$

Weighting factor of the progression rating *V* (section 9.1.5)

If the term “x” is specified, the global setting is used.

If **METHOD** ISO9790:

<g_V> Type: string

If the physical dimension of the x-axis is not time, then a different dimension can be specified alternatively. The codes of all allowed physical dimensions are described in ISO/TS 13499.

<g_P> Type: real, Domain: $\{ R_{>0}, x \}$

Weighting factor of the phase shift rating P (section 9.1.5)

If the term “x” is specified, the global setting is used.

<g_G> Type: real, Domain: $\{ R_{>0}, x \}$

Weighting factor of the size rating G (section 9.1.5)

If the term “x” is specified, the global setting is used.

<g_1> Type: real, Domain: $\{ R_{>0}, x \}$

Weighting factor of the corridor rating C_1 (section 9.1.4)

If the term “x” is specified, the global setting is used.

<g_2> Type: real, Domain: $\{ R_{>0}, x \}$

Weighting factor of the cross correlation rating C_2 (section 9.1.5)

If the term “x” is specified, the global setting is used.

<a_0> Type: real, Domain: $\{ [0,1], x \}$

Relative width of the inner corridor (section 9.1.4)

If the term “x” is specified, the global setting is used.

<b_0> Type: real, Domain: $\{ [0,1], x \}$

Specifies the relative width of the outer corridor (section 9.1.4)

If the term “x” is specified, the global setting is used.

<a_t> Type: string, Domain: {file name, NOTSPEC}

This parameter specifies the name of a file that contains corridor curves. Up to four curves can be specified in this file. Details are given in section 10. A curve requires at least two supporting points. The time unit is the output time unit specified in `output_units.txt`.

This function can be disabled by using the term “NOTSPEC”.

<a_s> Type: real, Domain: { [0,1] , x }

This parameter defines a scaling factor (a_s) to influence the inner corridor by means of the standard deviation (section 9.1.4).

If the term “x” is specified, the global setting is used.

<b_s> Type: real, Domain: { [0,1] , x }

This parameter defines a scaling factor (b_s) to influence the outer corridor by means of the standard deviation (section 9.1.4).

If the term “x” is specified, the global setting is used.

<D_min> Type: real, Domain: { [0,1] , x }

The parameter limits the allowed phase shift to calculate the phase shift rating (section 9.1.5). Contrary to the global definition, the value must be specified in the time unit of the outputs (section 12).

If the term “x” is specified, the global setting is used.

<D_max> Type: real, Domain: { [0,1] , x }

The parameter limits the allowed phase shift to calculate the phase shift rating (section 9.1.5). Contrary to the global definition, the value must be specified in the time unit of the outputs (section 12).

If the term “x” is specified, the global setting is used.

<Filter>

Type: character, Domain: {0, A, B, C, D}

This parameters selects the channel filter class to filter the signals prior any data processing (section 15). The following settings are accepted:

"0"	no filter applied
"A"	CFC1000
"B"	CFC600
"C"	CFC180
"D"	CFC60

9.2.4 Example of a load case parameter block

All parameters of the corridor method are printed in blue and the parameters of the cross correlation method are printed in red.

```
BEGIN LOADCASE
# loadcase 1
  NAM_LC          LC_1
  DES_LC          Description_LC1
  WF_LC           1.0
  PreT_LC         x
  PostT_LC        0.0
  METHOD           CORA

  BEGIN DATAFILES
#   Name          unit          g      time-shift
#   Test 1 to 3 will be used to calculate average test curves
  FileTest1       mm.K.s       YES    0.0
  FileTest2       m.g.s        YES    0.0
  FileTest3       mm.m.K.ms    YES    2.0
#   Simulation
  FileSimulation   m.K.s        No     -10.0
  END DATAFILES

  BEGIN SIGNALS
#   Signals
#   1 Name, 2 WF, 3 Y_norm, 4 t_min, 5 t_max, 6 g_V, 7 g_G, 8 g_P, 9 g1, 10 g2
#   11 a_0, 12 b_0, 13 a_t, 14 a_s, 15 b_s, 16 D_min, 17 D_max, 18 Filter
#   1          2      3      4      5      6      7      8      9      10     11     12     13      14     15 16 17 18
  S1HEAD0000H3ACXP 0.4 extremum x    0.1 x    x    x    x    x    0.1 1.0 NOTSPEC 0.3 1 x x A
  S1CHST0000H3DSXC 0.6 extremum 0.1 0.2 0.3 0.4 0.3 0.4 0.6 0.1 1.0 cor.dat 0 0 x x 0
  END SIGNALS
END LOADCASE
```

9.3 Sub-load case parameters

The parameters of load cases and sub-load cases are almost identical. Differences are shown in the following sections.

9.3.1 General Parameters

The load case keywords **NAM_LC**, **DES_LC** and **WF_LC** are replaced by specific sub-load case parameters. All other keywords and parameters remain the same.

NAM_SLC <string>

Name of the sub-load case (title)

DES_SLC <string>

Additional description of the sub-load case (sub-title)

WF_SLC <real> Domain: $R_{>0}$

Weighting factor of the sub-load case (section 4)

9.3.2 Data file block parameters

A sub-load case uses the same parameters as defined for a load case (section 9.2.2).

9.3.3 Signal block parameters

A sub-load case uses the same parameters as defined for a load case (section 9.2.3).

9.3.4 Example of a load case block with two sub-load cases

The following example uses the biofidelity rating method. Only the red signal parameters are taken into account when using this method.

```
BEGIN LOADCASE
  METHOD          ISO9790
  NAM_LC         LC_1
  DES_LC         Description_LC1
  WF_LC          1.0
  PreT_LC        0.0
  PostT_LC       0.0

  BEGIN SUBLOADCASE
    NAM_SLC       SLC_1
    DES_SLC       Description SLC 1
    WF_SLC        5.0

    BEGIN DATAFILES
      #      Name          unit      g      time-shift
```

CORApplus 4.1.0 User's Manual

```

    Test1.dat      mm.K.s      YES      0.0
END DATAFILES

BEGIN SIGNALS
#   Signals
    00SENS000000FOX0 8 corridor x x x x x x x x x corridor_1.dat x x x x D
    00SHLD000000DSY0 6 peak    x x x x x x x x x corridor_2.dat x x x x C
END SIGNALS

END SUBLOADCASE

BEGIN SUBLOADCASE
    NAM_SLC      SLC_2
    DES_SLC      Description SLC 2
    WF_SLC       5.0

    BEGIN DATAFILES
#   Name      time-shift      unit      g
    Test2.mme      0.0      mm.K.s      YES
    END DATAFILES

    BEGIN SIGNALS
#   Signals
    00ABD0000000DSY0 9 peak_low x x AC x x x x x x corridor_3.dat x x x x B
    END SIGNALS

END SUBLOADCASE

END LOADCASE

```

10 Corridor files

User-defined corridors can be specified in the signal parameters block by the parameter [a.t](#). Various ASCII format are supported (section 2.1). Depending on the selected metrics, a fixed number of curves in a defined order is required. The content is controlled by the keyword **METHOD** (section 9.2.1) and the signal parameter [Y_norm](#) (section 9.3.3).

The units of the corridor files are defined by the internal units file: [internal_units.txt](#).

10.1 Corridor method

If the parameter **METHOD** is set to "CORA", the following corridor curves shall be specified:

- lower inner corridor curve
- upper inner corridor curve
- lower outer corridor curve
- upper outer corridor curve

10.2 Biofidelity method

If the parameter **METHOD** is set to "ISO970", the number and order of the corridor curves depends on the settings of the signal parameter [Y_norm](#).

10.2.1 Corridor and Peak rating

If the signal parameter [Y_norm](#) is set to "corridor" or "peak", the following curves shall be specified:

- lower inner corridor curve
- upper inner corridor curve

The outer corridors are calculated automatically by adding and subtracting the maximum width of the inner corridors to the corridor curves.

10.2.2 Peak low/high rating

If the signal parameter [Y_norm](#) is set to "peak_high", "peak_low", "peak_high_5", or "peak_low_5" the following curve shall be specified:

- lower or upper limit curve

11 Configuration file

The configuration file `CORApplus.cfg` is written in ASCII format and is located in the sub-directory “config” of the software's installation directory. It contains several settings to build-up the HTML report. Comment lines starts with a hashmark (“#”).

11.1 Keywords

The order of the keywords is arbitrary.

BROWSER <file name>

This parameter specifies the command to start the preferred HTML viewer (browser). If CORApplus was started with the command line parameter “-b”, the HTML report (e.g. `result_CORApplus.html`) is immediately shown in the specified browser after the completion of the CORApplus analysis.

Example:

```
BROWSER    c:\Programme\Mozilla Firefox\firefox.exe
```

color <integer> Domain: $N_{>0}$

The parameter defines a colour by specifying name, pigment content value of the colours red, green, and blue and finally opacity. All values must range from “0” to “1”.

backgroundcolor <string>

The parameter sets the background colour of the charts shown in the HTML report. The colour must be pre-defined with the keyword **color**.

Example:

```
backgroundcolor    cyan
```

foregroundcolor <string>

The parameter sets the foreground colour of the charts shown in the HTML report. It is used to draw the frames of chart and legend as well as the axes. The colour must be pre-defined with the keyword **color**.

innercolor <string>

The parameter sets the colour of the inner corridor curves. The colour must be pre-defined with the keyword **color**.

outercolor <string>

The parameter sets the colour of the outer corridor curves. The colour must be pre-defined with the keyword **color**.

experimentcolor <string>

The parameter sets the colour of the reference curve. The colour must be pre-defined with the keyword **color**.

simulationcolor <string>

The parameter sets the colour of the comparison curve. The colour must be pre-defined with the keyword **color**.

intervalcolor <string>

The parameter sets the colour of two vertical lines that show the starting (t_{min}) and ending (t_{max}) time of the interval of evaluation. The colour must be pre-defined with the keyword **color**.

referencecolor <string>

The parameter sets the colour of the shifted reference curve that is the base of the cross correlation rating. The colour must be pre-defined with the keyword **color**.

Example:

color	cyan	0.00	1.00	1.00	1.00
referencecolor	cyna				

11.2 Obsolete Keywords

Previous releases of CORApplus used 3rd party tools to create the charts. This function is still available and can be enabled by using one of the following keywords:

HYPERGRAPH <file name>

The keyword specifies the command to start Altair HyperGraph to create the charts of the HTML report.

LSPOST <file name>

The keyword specifies the command to start LS-PrePost to create the charts of the HTML report.

PAMVIEW <file name>

The keyword specifies the command to start PAM-VIEW to create the charts of the HTML report.

Example:

```
HYPERGRAPH          c:\program files\altair\hg.exe
```

When using one of those keywords, the following parameters must be defined too.

CURVE_COLORS <string>

The keyword specifies the colours that have to be used by the third party software to create the charts.

Example:

```
CURVE_COLORS    6=magenta    5=cyan    7=yellow    8=crimson  
9=darkcyan
```

12 Definition of the system of units

The configuration file `output_units.txt` is using an ASCII format and it is located in the sub-directory “config” of the software's installation directory.

Every physical dimension, used by a CORApplus evaluation, must be assigned to a physical unit. The used dimension code shall comply with ISO/TS 13499 (13th and 14th character of the ISO channel name). The units are described by their SI description. It is possible to use equations of basic units to describe derived units. However, it is not allowed to use derived units to describe other units.

A copy of `output_units.txt` with the most common physical dimensions and their corresponding unit is delivered with the CORApplus software package. So there is usually only a need to edit this file, if different units are required for the HTML report (e.g. [mm] instead of [m]).

12.1 Syntax

Comment lines shall start with a hashmark (“#”). It is also possible to add a comment at the end of a line. Those in-line comments start with a hashmark too.

A physical dimension is defined by two capital letters according to [3], separated by a semi-colon (“;”) from its unit.

Example:

```
# Comment
```

```
AC; m/(s*s)          # comment
```

12.2 Basic units

The following basic units are supported by CORApplus:

Time

second [s] and millisecond [ms]

Mass

kilogram [kg] and ton [t]

Length

metre [m] and millimetre [mm]

In addition, the unit [g] is supported for linear accelerations (1 g=9.81 m/s²).

12.3 Derived units

Only units, derived from the basic units of time, mass, and length (section 12.2) can be converted from the system of units of a data file to that of the corresponding outputs (HTML report). The following examples demonstrate the conversion of units:

Example of supported conversions:

Power P0

P0; W # Watt

P0; mW # Milliwatt

P0; (kg*m*m)/(s*s*s) # equivalent to [W]

Example of not supported conversion:

Power P0

P0; N*m/s # [N] is a derived unit (N=(kg*m)/(s*s))

P0; kg*m/s^3 # character "^" not supported, use (s*s*s)

P0; (kgm)/(s*s*s) # missing character "*", use (kg*m)

P0; (Kg*m)/(s*s*s) # case sensitivity error, use (kg*m)

12.4 Units in the ASCII control file

The units in the control file are defined by the configuration file `internal_units.txt`.

It has the same format and content of the `output_units.txt` file. This file defines the

unit of `t_min`, `t_max`, `Y_NORM` and corridor curves from file formats without a unit. This is a change to Cora Versions < 4.0 which expected these data to use the output system of units.

12.5 Unit mapping file

With Version 4.0.2 a unit mapping file was introduced. The file is called "map_units.txt". The file can be used to define units which CORApplus does not understand. The format is

XX; unknown unit, known unit

or

XX; unknown unit, factor to SI unit

XX is the ISO-MME short form for a signal type like AC = acceleration.

The "*unknown unit*" is a unit string CORApplus does not know.

The "*known unit*" is the same definition in a format CORApplus does know.

If such a conversion is not possible it is also possible to define the factor to the SI unit. For example: CORApplus does not support inch=2.54cm=0.0254m. So it would be possible to use inch like this:

DS; inch, 0.0254

The usage of the file map_units.txt is optional.

13 Signal name mapping file

Generally, it is recommended to use signal names that comply with ISO/TS 13499 coding. However, CORApplus offers the possibility to use arbitrary signal names. If so, a name mapping file must be prepared that associates those names with ISO/TS 13499 names. This ASCII file ("name_mapping.txt") shall either be located in the working directory or in the "config" directory of the software's installation.

Comment lines starts with a hasmark ("#"). Each name mapping line must start with the ISO name followed by alternative names. All names must be separated by a semicolon (";").

Example:

Comment

S1HEAD0000H3ACXA; X-Acceleration; H350_Dummy-Head-Acc-X

The setting of the global keyword **ISONAME_1-2/11-12** (section 9.1.1) shall be considered when using the name mapping function.

Example:

ISONAME_1-2/11-12 YES YES

The signal name **S1**HEAD0000**H3**ACXA is equivalent to **00**HEAD0000**HF**ACXA and the other way round.

14 Program flow

This chapter describes the program flow in detail.

- (1) Analysis of the command line parameters
- (2) Reading the configuration file (config.txt) and the signal name mapping file (name_mapping.txt)
- (3) Reading the control file
 - (3.1) Global parameters

At first, the software searched for the keyword **BEGIN GLOBAL_PARAMETERS**. The subsequent lines must contain all global parameters (section 9.1). This block ends with the keyword **END GLOBAL_PARAMETERS**.

- (3.2) Load cases

A load case block starts with the keyword **BEGIN LOADCASE** and ends with **END LOADCASE**. The definition of a load case must start with the general load case parameters (section 9.2.1) followed by the data files block (section 9.2.2) and the signal parameters block (section 9.2.3).

The data file block starts with **BEGIN DATAFILES** and ends with **END DATAFILES**. Each line of this section must comply with the definitions described in section 9.2.2. The first up to the second of last line defines the reference data. If more than one reference data file is specified, average reference data are calculated. The last line of this section specifies always the comparison data. Therefore, at least two data files (two lines) must be specified within this block.

The signal parameters block starts with **BEGIN SIGNALS** and ends with **END SIGNALS**. The signal block selects the curves of the data files that have to be evaluated. All curves should use the name conventions according to ISO/TS 13499. These names are used to identify the curves in the data files. Alternative names, defined in the name mapping file are consid-

ered too. All signal parameters (section 9.2.3) must be specified in the defined order, even if they are not used.

Any number of additional load cases can be defined after the keyword **END LOADCASE**.

(3.3) Sub-load cases (if existing only)

A sub-load case adds a new hierarchy level to the rating. A load case which has sub-load cases cannot contain signals.

A sub-load case starts with **BEGIN SUBLOADCASE** and ends with **END SUBLOADCASE**. It must contain some general parameters (section 9.3.1), a data file block (between **BEGIN DATAFILES** and **END DATAFILES**, section 9.3.2) and a signal parameters block (between **BEGIN SIGNALS** and **END SIGNALS**, section 9.3.3). The syntax of these blocks is identical to that of a load case.

Any number of additional sub-load cases can be defined after the keyword **END SUBLOADCASE**.

(4) Check of the weighting factors

The weighting factors of signals, sub-load cases, and load cases are normalised. More information about the weighting factors is given in section 4.

(5) Reading the curves

The format of the curve files (data files) is automatically detected. CORApplus identifies all curves of a data file but only read those that are specified in the signal definition block.

The following data processing is applied to the read curves:

- (5.1) Filtering (if specified)
- (5.2) Conversion of the physical units (see `output_units.txt`)
- (5.3) Shifting of the curve along the x-axis (if specified)
- (5.4) Re-sampling

CORApplus requires curves of the same sampling rate. Therefore, the minimum sampling rate of all specified curves of a signal is calculated. This rate is increased to an integral number of supporting points within the interval of evaluation if this interval is set manually. In case of using the automatically calculated interval of evaluation, the starting and ending time of the whole signal is used instead.

All curves are interpolated by a cubic spline function (section 16).

(5.5) Calculation of the mean reference curve

A mean reference curve $x(t)$ is calculated if more than two reference data files are specified in the data files block of a load case or sub-load case.

$$x(t) = \frac{\sum_{i=1}^{n_{Test}} x_i(t)}{n_{Test}}$$

(6) Generation of the corridor curves (CORA method only and [a_t](#) set to "NOTSPEC")

The inner and outer corridor is generated around the only reference curve or the mean reference curve. The values defined by global keywords ([Y_NORM](#), [a_0/b_0](#), [a_sigma/b_sigma](#)) and local parameters ([Y_norm](#), [a_0](#), [b_0](#), [a_s](#), [b_s](#)) are used.

The calculation of Y_{norm} depends on the interval of evaluation. If both bounds are calculated automatically, then Y_{norm} is calculated for the entire reference curve (between starting and ending time of the curve). Otherwise it is calculated for that part of the curve that is not excluded by the user-defined bounds (e.g. starting time to t_{max} , if t_{max} is given).

(7) Definition of the interval of evaluation (CORA method only)

The bounds of the interval can be defined globally ([T_MIN/T_MAX](#)) or locally ([t_min](#), [t_max](#)). If the term "automatic" is specified, the corresponding limit is calculated by using the algorithms of CORApplus (section 3.2).

CORApplus checks, if the interval of evaluation exceeds the domain of the reference curve $x(t)$ or the evaluated curve $y(t)$. If so, the interval will be shortened automatically.

When the automatic algorithm is used, the spline interpolation is done with the whole curves a second time to get a pair of values at the begin and end of the interval.

(8) Small signals (CORA method only)

If the global parameter **MIN_NORM** is not equal "0", a special treatment for small signals is activated (section 9.1.3).

(9) Reading the corridor files to be used with the CORA corridor method (CORA method: parameter **a_t** used) or the biofidelity rating (ISO9790 method only)

(10) Corridor correlation (CORA method only)

The rating C_1 according to the corridor metric is calculated for each signal (section 5).

(11) Cross correlation (CORA method only)

The rating C_2 according to the cross correlation metric is calculated for each signal (section 6).

(12) Total signal rating

The total rating C_3 is calculated for each signal (section 4).

(13) Signal rating (ISO9790 method only)

The biofidelity rating C_3 is calculated (section 7).

(14) Calculation of the rating of sub-load cases, load cases and the final total rating

(15) Output of the ASCII result file

CORApplus exports the processed files (curves, corridors etc.).

(16) Output of the HTML report

15 CFC Filter

In accordance with the SAE J211 [4], a 4-channel Butterworth low pass filter can be applied to the signals prior any other signal processing.

$$Y(t) = a_0 \cdot X(t) + a_1 \cdot X(t-1) + a_2 \cdot X(t-2) + b_1 \cdot X(t+1) + b_2 \cdot X(t+2)$$

$Y(t)$ is the filtered curve (output) and $X(t)$ the original curve (input). The filter constants a and b are calculated by using the following equations:

$$a_0 = \frac{\omega_a^2}{1 + \sqrt{2 \cdot \omega_a + \omega_a^2}}$$

$$a_1 = 2 \cdot a_0$$

$$a_2 = a_0$$

$$b_1 = \frac{-2 \cdot (\omega_a^2 - 1)}{1 + \sqrt{2 \cdot \omega_a + 2 \cdot \omega_a^2}}$$

$$b_2 = \frac{-1 + \sqrt{2 \cdot \omega_a - \omega_a^2}}{1 + \sqrt{2 \cdot \omega_a + \omega_a^2}}$$

ω_a is derived from ω_d and T is the sampling rate in seconds. To prevent phase shifts, the filtering has to run twice – forward and backward.

$$\omega_a = \frac{\sin \omega_d \frac{T}{2}}{\cos \omega_d \frac{T}{2}}$$

$$\omega_d = 2 \cdot \pi \cdot CFC \cdot 2.0775$$

16 Spline interpolation

All curves are interpolated by harmonic cubic splines to get a consistent sampling rate. A brief summary of the methodology is given in this section. Further information is available in [1]. A cubic spline for $n+2$ points $((x_0, y_0), (x_1, y_1), \dots, (x_{n+1}, y_{n+1}))$ must be chosen, resulting in the following equation:

$$\begin{pmatrix} 2 & \lambda_0 & 0 & 0 & 0 & 0 & \dots \\ \mu_1 & 2 & \lambda_2 & 0 & 0 & 0 & \dots \\ 0 & \mu_2 & 2 & \lambda_2 & 0 & 0 & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & 0 & \mu_n & 2 & \lambda_n \\ 0 & 0 & 0 & 0 & 0 & \mu_{n+1} & 2 \end{pmatrix} \cdot \begin{pmatrix} M_0 \\ M_1 \\ M_2 \\ \vdots \\ M_n \\ M_{n+1} \end{pmatrix} = \begin{pmatrix} r_0 \\ r_1 \\ r_2 \\ \vdots \\ r_n \\ r_{n+1} \end{pmatrix}$$

With the following parameters, the equation can be solved by inverting the matrix and calculating the M values:

$$\Delta x_i = x_{i+1} - x_i \quad 0 \leq i \leq n$$

$$\Delta y_i = y_{i+1} - y_i \quad 0 \leq i \leq n$$

$$\lambda_i = \frac{\Delta x_i}{\Delta x_{i-1} + \Delta x_i} \quad 1 \leq i \leq n \text{ and } \lambda_0 = 0$$

$$\mu_i = 1 - \lambda_i \quad 1 \leq i \leq n \text{ and } \mu_{n+1} = 0$$

$$r_i = \frac{6}{\Delta x_{i-1} + \Delta x_i} \cdot \left(\frac{\Delta y_i}{\Delta x_i} - \frac{\Delta y_{i-1}}{\Delta x_{i-1}} \right) \quad 1 \leq i \leq n, \quad r_0 = 0, \text{ and } r_{n+1} = 0$$

The coefficients a , b , c , and d of the cubic polynomials $s_i(x)$ can be calculated according to the following equations:

$$s_i(x) = d_i + c_i(x - x_i) + b_i(x - x_i)^2 + a_i(x - x_i)^3 \quad 0 \leq i \leq n$$

$$a_i = \frac{1}{6} \frac{(M_{i+1} - M_i)}{\Delta x_i}$$

$$b_i = \frac{1}{2} M_i$$

$$c_i = \frac{\Delta y_i}{\Delta x_i} - \frac{1}{6} \Delta x_i (2M_i + M_{i+1})$$

$$d_i = y_i$$

17 Encryption of the global parameters

CORApplus offers the possibility to protect the global parameters from any modification by encryption. Encryption is not introduced to hide the global parameters but to ensure that these parameters are used. The file `result_CORApplus.enc` is written automatically at the end of an evaluation when encrypted parameters are used. It contains the results (see also `result_CORApplus.txt`) and all global parameters that have been used. This file is encrypted as well.

The global parameters must be sourced out to a separate ASCII file that shall be encrypted afterwards. A password shall be specified to encrypt the file. No password is required to use encrypted parameters in a CORApplus evaluation.

17.1 Encryption

An additional tool (`CORApplus_encrypt`), delivered with the CORApplus software package, shall be used to encrypt the file with the outsourced global parameters. It is a command line tool, available for Linux and MS-Windows computers.

```
CORApplus_encrypt -e <file name1> <file name2> <password>
```

<file name1>

ASCII file with the global parameters (input)

<file name2>

Encrypted binary file (output)

<password>

Password to encrypt the file. It can be of arbitrary length and can contain any character except blanks and characters which are not printable.

17.2 Decryption

The tool `CORApplus_encrypt` cannot be used to decrypt an encrypted parameter file but it can be used to decrypt encrypted results, written in `result_CORApplus.enc`.

CORApplus_encrypt -d <file name1> <file name2> <password>

<file name1>

Binary file with the encrypted results (input)

<file name2>

ASCII file with the decrypted CORApplus results and the used global parameters (output)

<password>

The password, used to encrypt the global parameters, must be specified

17.3 Usage of encrypted parameters

When using the option to protect the global parameters, only the parameters shall be sourced out. The keywords that describe the global parameter block (**BEGIN GLOBAL PARAMETERS** and **END GLOBAL PARAMETERS**) still remain in the control file. The encrypted parameters are included by the keyword **PARAM_FILE**.

PARAM_FILE <file name>

The parameter specifies a file with encrypted global parameters. If this keyword is specified, no other global parameter is allowed in the control file.

17.4 Example

17.4.1 Encrypted parameter file

An example of an ASCII file with the global parameters is shown below. This file must be encrypted (section 17.1) before use.

```
# Global parameter block
A_EVAL                0.00
A_THRES                0.05
B_THRES                0.07
B_DELTA_END           0.1
T_MIN/T_MAX           automatic    1.0
Y_NORM                extremum
DES_GLO                what it is
DES_MOD               Model 42
FONT_SMALL            13
FONT_LARGE            16
ISONAME_1-2/11-12     YES      NO
OUTPUT_FORMAT         LSPOST
PreT_LC/PostT_LC      0.0      0.1
```

```

WF_NORM          YES
MIN_NORM         0.20
a_0/b_0          0.7   0.9
a_sigma/b_sigma  0.2   0.5
K                2
G_1              0.50
D_MIN            0.05
D_MAX            0.10
INT_MIN          0.3
K_V              1.0
K_G              1.0
K_P              1.0
G_V              0.33
G_G              0.33
G_P              0.33
G_2              0.50
DIM_WEIGHTS      AC    0.4   0.4   0.2   0.55  0.45
END GLOBAL_PARAMETERS

```

17.4.2 Control file

This example shows how to include an encrypted parameter file in a control file.

```

BEGIN GLOBAL_PARAMETERS
# encrypted parameter file
  PARAM_FILE      test.enc
END GLOBAL_PARAMETERS

```

18 Proposed settings

Based on the experiences with CORApplus over the past years, the following global settings are recommended for the use with responses of crash test dummies. However, other settings are possible and the user is encouraged to tune them to his own needs.

18.1 Interval of evaluation

<i>A_THRES</i>	0.030
<i>B_THRES</i>	0.075
<i>A_EVAL</i>	0.010
<i>B_DELTA_END</i>	0.200
<i>T_MIN/T_MAX</i>	automatic automatic

18.2 Corridor method

<i>Y_NORM</i>	extremum
<i>K</i>	2
<i>G_1</i>	0.50
<i>a_0/b_0</i>	0.05 0.50
<i>a_sigma/b_sigma</i>	0 0

18.3 Cross correlation method

<i>D_MIN</i>	0.01
<i>D_MAX</i>	0.12
<i>INT_MIN</i>	0.80
<i>K_V</i>	10

<i>K_P</i>	1
<i>K_G</i>	1
<i>G_V</i>	0.50
<i>G_G</i>	0.25
<i>G_P</i>	0.25
<i>G_2</i>	0.50

18.4 HTML report

<i>FONT_SMALL</i>	12
<i>FONT_LARGE</i>	14

19 References

- [1] Feldmann, D.: „Repetitorium der Ingenieur-Mathematik Teil 2“, Binomi Verlag GbR, Barsinghausen, Germany, 1993
- [2] “ISO/TR 9790: Road Vehicles – Anthropomorphic side impact dummy – lateral impact response requirements to assess the biofidelity of the dummy”, International Organization for Standardization (ISO), Geneva, Switzerland, 1999
- [3] “ISO/TS 13499: Road Vehicles – Multimedia exchange format for impact tests”, International Organization for Standardization (ISO), Geneva, Switzerland, 2002
- [4] “SAE J2111/1: “Instrumentation for Impact Test – Part 1 – Electronic Instrumentation”; SAE International, Troy MI, USA, 2007
- [5] “ISO/TS 18671: Road vehicles — Objective rating metric for non-ambiguous signals
- [6] Müller, Meinard: Information Retrieval for Music and Motion, 2007, SBN 978-3-540-74048-3

20 Release notes

20.1 CORApplus 4.1.0

- ISO-18571 rework:
 1. DTW path calculation problem with equal values in cost matrix fixed
 2. Slope averaging is now done by moving average calculation
 3. Phase shift score used full (integer) intervals

20.2 CORApplus 4.0.5

- Bugfix problem with closing files

20.3 CORApplus 4.0.4

- Renamed to CORApplus

20.4 Cora 4.0.3

- Bugfix
Fixed problem with unit mapping file
ISO-18571 corridors are not adjusted to (test) curves

20.5 Cora 4.0.2

Features

- Introduction of a unit mapping file: map_units.txt

Bugfix

- Removed crash on MS-Windows if the browser could not be found
- If K – values are not integer the program quits with an error message
- improved error messages

20.6 CORA 4.0

Features

- Rating according to ISO18571
- `internal_units.txt` file defines system of units of input data. This makes `T_UNIT` obsolete.
- The order of command line arguments is arbitrary

Change

- The order of the parameters after the curve file specification in the control file is now timeshift, units and acceleration in g. The last two options can be omitted for ISO-MME and Diadem files.
- ISO-MME and Diadem curve files define the units of the containing curves.

Bugfix

- Narrow corridor curves lead to wrong corridor ratings.

20.7 CORA 3.6.1

Bugfix

- Selection of the signal filter class in sub-load cases

20.8 CORA 3.6

New

- Keyword `T_UNIT` to select the unit of `T_MIN/T_MAX`, `t_min`, and `t_max` (section 9.1.1)
- The shifted signal to calculate the cross correlation is now shown in the HTML report

Change

- Calculation of the time shift of the cross correlation method → `INT_MIN` (section 6.3); WARNING: this change may affect the rating results; however, the rating is less sensitive against the total length of the signals

- A warning is shown if the time shift is limited by the length of the signal

Bugfix

- Handling of ISO-MME files