

Order analysis in ArtemiS SUITE¹

Introduction	1
Requirement: RPM information	3
Calculating an <i>Order Spectrum vs. RPM</i> with the <i>Variable DFT Size</i> algorithm	4
Basics	4
Configuration possibilities in the Properties window of the analysis	5
Order cuts and average order analysis	8
Order cuts	8
Averaged Order Analysis	9
Calculating an order spectrum with the <i>RPM-Synchronous Resampling</i> algorithm	10
Calculating an order spectrum with the <i>Time Domain Averaging</i> method	11
Choosing the best Settings	12
Note	12

Introduction

In the analysis of engine sounds, the rotational speed (revolutions per minute, RPM) plays an important role: Certain sound emissions repeat with each revolution at a specific angle of rotation, which results in a spectral content that corresponds to the multiples of the rotational frequency.

Frequencies that correspond to the motor's RPM or multiples of it are called orders. The first order is identical with the frequency of the RPM; the second order is the frequency of the first order multiplied by the factor 2 etc. Order analysis is the process for determining the signal level or level curve for specific orders.

The levels in an order analysis can be displayed in different ways: the averaged level of different orders (figure 1a) or a level curve versus time (figure 1b/1d) or versus RPM (figure 1c/1e). Furthermore, the results can be displayed either in an order spectrum (figure 1d/1e) or as a level curve of individual orders (figure 1b/1c).

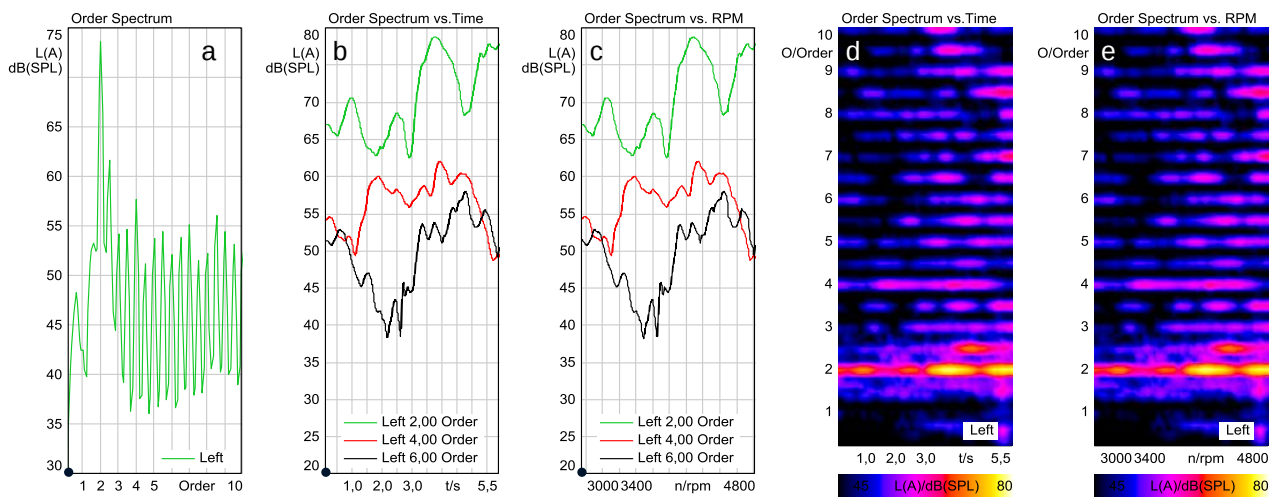


Figure 1: Different variants of order analysis

Figure 2 shows an order analysis compared to an **FFT vs. Time** and an **FFT vs. RPM** analysis. The **FFT vs. Time** analysis on the left shows the level of a sound file as a function of time (x-axis) and also as a function of frequency (y-axis).

¹ The descriptions in this Application Note refer to version 8.0. The general procedures also apply to other versions. However, the scope of functionality and the user interface may differ.

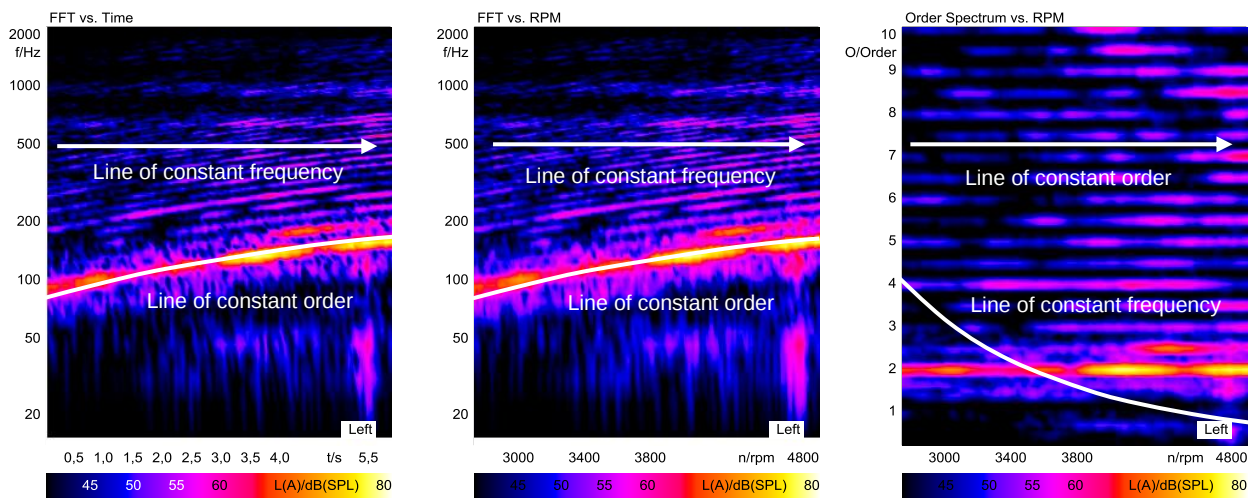


Figure 2: FFT vs. Time, FFT vs. RPM and Order Spectrum vs RPM

The color coding of the spectrum allows the viewer to determine the sound level from the diagram both versus time and versus frequency.

In the **FFT vs. RPM** analysis (center), the x-axis is used for RPM instead of time. That way the viewer can determine the sound level spectrum, which is generated at any given motor speed. Since the RPM of the sound file, used for this example, increases linearly with time, the results of the **FFT vs. Time** analysis do not differ very much from those of the **FFT vs. RPM** analysis.

In the **Order Spectrum vs. RPM** analysis on the right, the x-axis is used for RPM, as it is in the **FFT vs. RPM** analysis. However, the y-axis no longer represents the frequency values in Hz, but instead the rotational frequency and its multiples, i.e. the orders. To achieve this, the frequency axis changes depending on the current RPM, so the orders no longer appear as curves, but as straight lines in the diagram. The diagram thus shows the dependency of the sound level on both the revolution speed and the order.

The following sections contain a detailed description of the procedure used to calculate an order analysis. First, the calculation of an order spectrum versus revolution speed using the order algorithm **Variable DFT Size** is described in detail and the various settings in the Properties window of the analysis are explained. Based on this representation, the calculation of the level curves of individual orders (cut function) and the averaged level analysis are described. Afterwards, the order algorithms **RPM-sync. Resampling** and **Time Domain Averaging** are explained. At the end, the notes on applying the analyses are summarized.

Requirement: RPM information

As described above, an order analysis requires RPM information. This information can either be stored in a digital pulse channel or in an additional, analog channel. Depending on how you recorded your RPM information, it may be necessary to decode it first, e.g., if the RPM data was provided via a CAN bus for the measurement. For this purpose, you can use a Decoder Project in ArtemiS SUITE.

Once the RPM information is available in a form that can be used in ArtemiS SUITE, you can select the channel containing the RPM information in the Properties window of the sound file (figure 3).

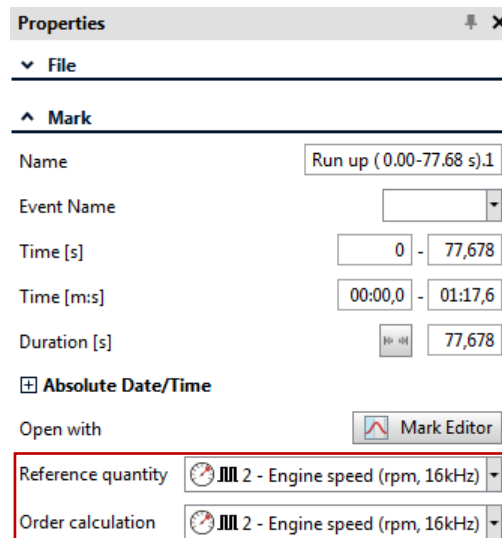


Figure 3: Properties dialog of a sound file

The **Reference quantity** field is used for selecting the channel containing the reference quantity against which the analysis result is to be plotted. The **Order Calculation** field specifies the channel containing the reference quantity from which the orders are to be determined by the order analysis. In most cases, the same channel is selected for both fields. ²

If you were unable to record RPM information when measuring your signals or if your RPM measurement is corrupted, you can use the RPM Generator of ArtemiS SUITE to derive an artificial RPM curve from the order curves of the signals and save it.

² For more detailed information on how RPM information can be pre-processed in ArtemiS SUITE and how the selected reference quantity and order calculation affect your analysis results, please refer to the Application Notes "Pre-processing of revolution speed data" and "Using different reference quantities".

Calculating an *Order Spectrum vs. RPM* with the *Variable DFT Size* algorithm

Basics

For an RPM-dependent order analysis, the time-domain signal is not analyzed continuously, but instead selectively at certain positions separated by constant revolution speed intervals. You can define these intervals in the Properties window of the order analysis (see next chapter “Properties window of the analysis”).

At these positions, a time interval ΔT , located symmetrically around each point, is analyzed with a DFT (Discrete Fourier Transform). If a step width of 50 RPM is selected for the analysis of a measurement that was started at 920 RPM, the first DFT analysis is performed at 950 RPM, provided that a time window of the length ΔT can be positioned around this value. The next sampling point is then at 1000 RPM etc. Once the first sampling point has been found, all further DFTs are performed in the specified RPM step width. If a DFT has been calculated at 1500 RPM and the step width is set to 100 RPM, the next point in time for an analysis to be made is when the rotational speed is 1600 RPM. The exact time selected for the analysis is the point where the revolution speed reaches the value 1600 for the first time. If the RPMs are not constantly increasing (or decreasing) and the value 1600 is reached again, this point will be ignored and only the first point where this RPM value is reached is used for the DFT analysis.

Figure 4 on the next page schematically shows the calculation procedure of an ***Order Spectrum vs. RPM*** using the order algorithm ***Variable DFT Size***. First, DFT analyses with the RPM-dependent window size ΔT are performed at the supporting points of the RPM curve determined by the specified step size. In this example, the revolution speed increases, i.e. the width of the analysis window gets smaller with each step.

The analysis results of all the RPM sampling points are finally displayed in a three-dimensional diagram. The result displayed by ArtemiS SUITE is a spectrogram, where the level values are color-coded. The x-axis of this diagram represents the revolution speed, and the y-axis represents multiples of the revolution speed frequency (orders).

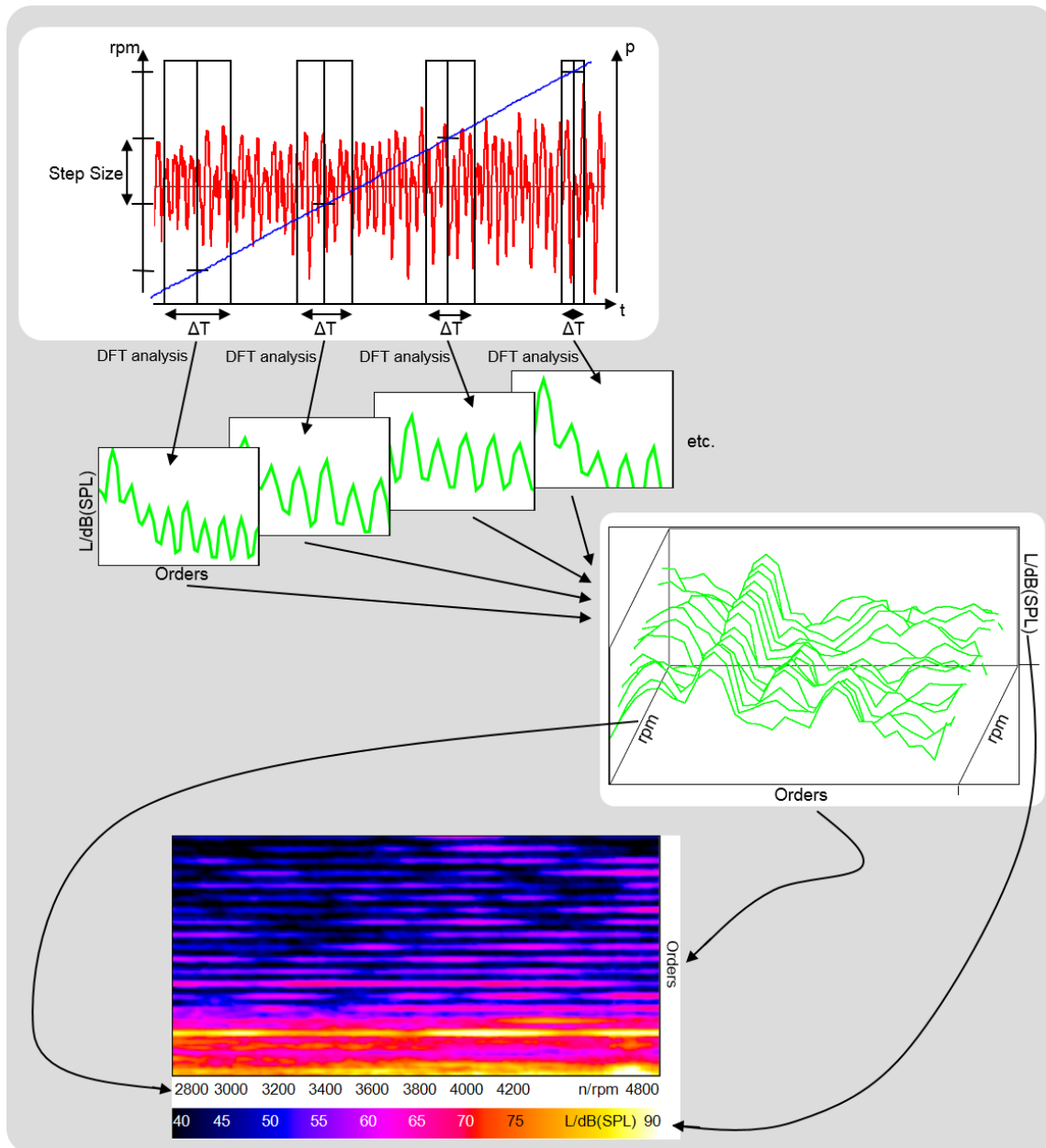


Figure 4: Scheme of the *Order Spectrum vs. RPM* analysis

Configuration possibilities in the Properties window of the analysis

Figure 5 shows the Properties window of the *Order Spectrum vs. RPM* analysis. In the **Step Size** field, you can specify the distance between the RPM values for the DFT analyses mentioned above. In the **Slope** select box, the direction of the RPM change can be specified, e.g. **Rising** for an engine run-up or **Slope Detection** if the direction of the RPM change is to be detected automatically.

Order Spectrum vs. RPM

Window Function	Hanning
Spectral Weighting	None
Amplitude Scaling	RMS
Frequency Offset [Hz]	0
Spectral Resolution [Order]	0,5
Width Definition	Off
Spectral Range [Order]	0 to 20
Phase ref. to	Off
Order Algorithm	Variable DFT Size
Step Size [rpm, ...]	50 Slope Detection
Time Weighting	Manual
Time Constant [ms]	0

Figure 5: Properties window of the **Order Spectrum vs. RPM** analysis

The order resolution is specified via the **Spectral Resolution [Order]** value. When using the calculation algorithm **Variable DFT Size**, this value determines the current window width and time resolution ΔT of the DFT analysis depending on the current revolution speed. The following formula represents this dependency:

$$\Delta T = \frac{60}{\text{rpm} \cdot \text{order resolution}}$$

The formula shows that in this calculation algorithm, the time window width (Fourier analysis block length) depends on the RPM. Therefore the window width decreases with increasing RPM values. Furthermore, the window width is inversely proportional to order resolution. This means, the finer the selected resolution, the longer the analysis time window gets. At a resolution of 0.1, each time window ΔT covers 10 revolutions, whereas at an order resolution of 0.5 it is only 2 revolutions. Similar to an FFT analysis, which is subject to an effect called frequency-time uncertainty, a higher time resolution of the order analysis leads to a lower order resolution and vice versa. Figure 6 shows the difference between an order analysis with an order resolution of 0.01 and one with a resolution of 0.2. All other parameters have been kept identical.

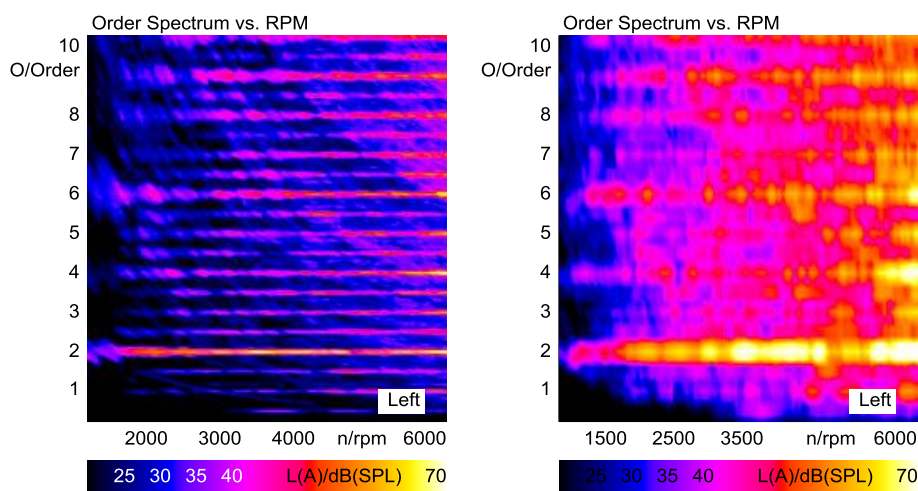


Figure 6: Comparison of different order resolutions; left diagram: order resolution 0.01 right diagram: order resolution 0.2

If you want the width of the displayed orders to differ from the order resolution, you can sum up the signal powers of the individual DFT lines in broader bands by changing the value in the **Width Definition**

field. This can be particularly useful if orders appear “smeared” when using a high order resolution, e.g. due to revolution speed fluctuations in the in the DFT window. The following settings are available³:

- **Off**: With this option, the DFT lines will not be integrated. The width of the orders is therefore equal to the value specified in the **Spectral Resolution** field.
- **Order**: If the **Order** option is used, the width in orders can be specified in the field **Width**. The DFT lines are integrated according to this width, which is symmetrically split around the nodes. Thus, high resolution and reasonable bandwidth can be combined in order to e.g. determine the levels of fractional orders with an appropriate width.
- **Frequency**: With this option, the integration is done with a fixed bandwidth, which is specified in Hz in the **Width** field. The higher the speed of rotation is, the narrower the width (measured in order) gets.
- **Frequency Factor**: With this option, the bandwidth of integration is defined by a factor. Seen on a logarithmic frequency scale, the integration bandwidth is symmetrical around the respective order, e.g. “0.707” ($= \sqrt{2}/2 = \text{octave width}$) can be a reasonable factor.
- **Bark**: The **Bark** option effects that the DFT lines are integrated in critical bandwidths according to the value entered in the field **Width**.

If the selected width is lower than the selected spectral resolution, the value of the spectral resolution defines the analysis. (This corresponds to the option **Off**).

For the analysis window used for the DFT, several parameters can be specified in the Properties window. These include the desired **Window Function**, e.g. **Rectangle**, **Hanning**, **Hamming**, or **Kaiser-Bessel**, and parameters for **Time Weighting** and **Spectral Weighting**.

The **Time Weighting** setting specifies the integration time period, over which the level curves are to be averaged. Depending on the selected integration time, this leads to more or less smoothing of the level curve. The **Spectral Weighting** setting allows the level curve to be displayed with A-, B-, C- or D-weighting. The difference between an A-weighted and an unweighted order analysis is shown in figure 7. By using A-weighting the level is reduced at high and low frequencies. Due to the selection of the displayed axis range, the decay at low frequencies can be seen particularly well in figure 7.

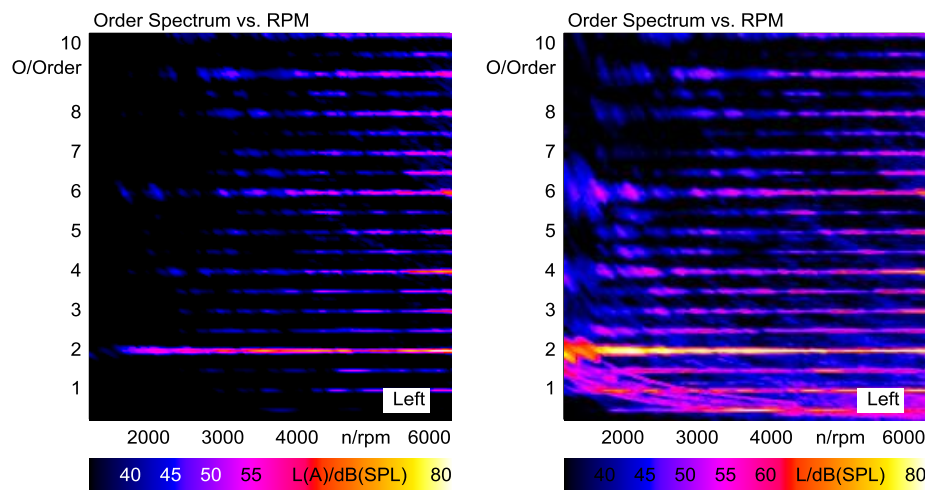


Figure 7: Comparison of an A-weighted order analysis (left diagram) and an unweighted order analysis (right diagram)

In the **Amplitude scaling** field, the scaling can be switched between **RMS** and **Peak**. The first setting displays the effective value (root mean square) of the oscillation, whereas the second setting displays the peak value, which is greater than the RMS value by a factor of $\sqrt{2}$ in case of a sine wave.

³ In an **Order Spectrum** analysis, a width setting other than **OFF** can cause the calculated single-value result to be considerably greater than the actual signal power.

By entering a **Frequency Offset**, you can add a constant value to the order frequency. This is advisable in cases where the frequency curve not only depends on the revolution speed, but also on a fixed frequency.⁴

The numbers specified under **Spectral Range** define the analysis range, i.e. the minimum and maximum order to be analyzed. Furthermore, the **Phase** select box provides a choice whether the result should include complex data. If **Off** is selected, only the level of the orders is calculated and displayed. With the settings **Channel**, **Order**, or **Pulse**, a complex order spectrum is calculated relative to the selected reference channel, the selected order, or the selected pulse channel, respectively.⁵

In the selection box **Order Algorithm**, you can select the desired algorithm for calculating the analysis: **Variable DFT Size**, **RPM-synch. Resampling**, or **Time Domain Averaging**.

Of course, the order analysis cannot only be calculated and displayed versus the revolution speed, but also versus time. The Properties window of this analysis widely contains the same parameters as the one shown and explained above, but in this case, the results are not plotted against an RPM axis, but against a time axis. Therefore, the **Step Size** parameter is not entered in RPM as for the **Order Spectrum vs. RPM**, but in milliseconds.

Order cuts and averaged order analysis

Order cuts

The cut function can be used to display the level curve(s) of one or several orders separately. This function can be enabled or disabled in the **Cuts** section in the Properties window of the analyses **Order Spectrum vs. Time / RPM**. The default setting cuts the 2nd, 4th and 6th order from the spectrum. The resulting cuts then show the level curve of the respective order versus time or RPM in a 2D diagram (see figure 8 on the next page).

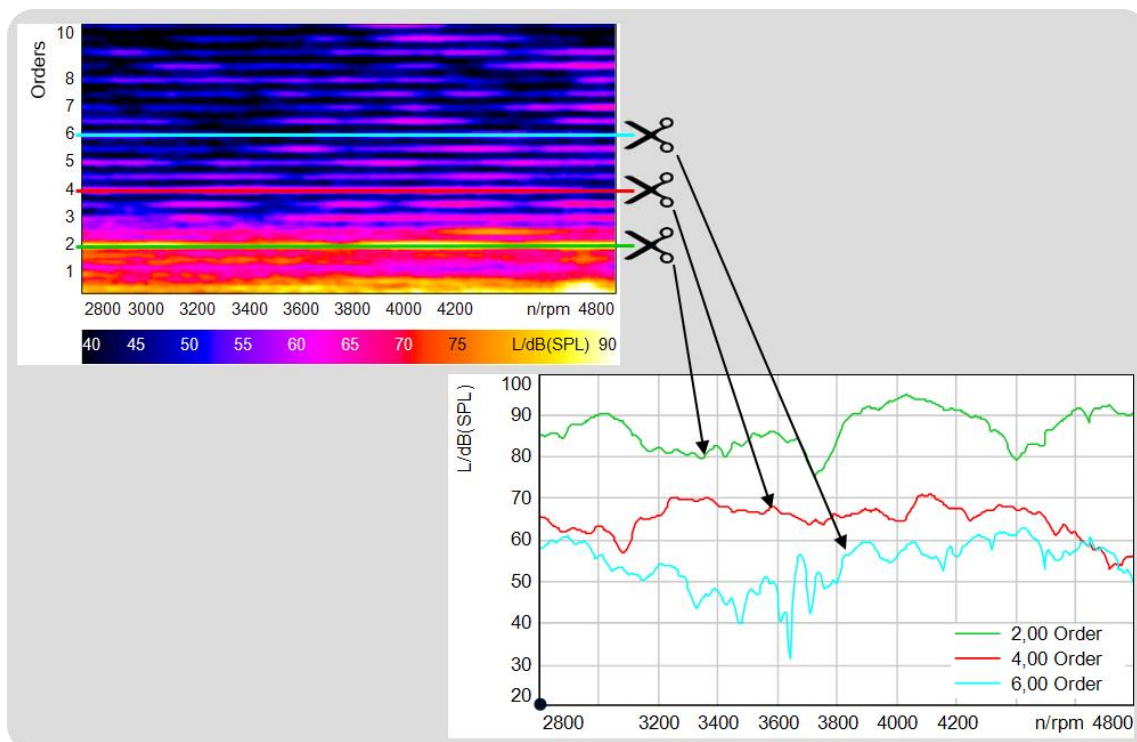


Figure 8: Order cut from a spectrogram

⁴ A detailed description of this function can be found in the Application Note "Using a Frequency Offset in Order Calculations".

⁵ For the phase to be displayed in a diagram, the representation of complex numbers must be enabled in the diagram settings.

The underlying spectrogram is calculated according to the analysis parameters configured in the Properties window. Therefore, the displayed curves may differ depending on the configured analysis parameters. In order to obtain comparable results from different calculations, make sure to use identical parameter sets for all calculations.

Averaged Order Analysis

Furthermore, it is possible to display a level curve averaged over time or RPM. The analysis result displayed in this case is the level of the different orders averaged over all time or RPM points. In the Properties window, you can select in the field **Average Mode** whether a diagram versus time or a diagram versus RPM should be used as the basis for averaging. Depending on this choice, the distance between the individual analysis points is given either in RPM or in milliseconds. For a nearly steady RPM situation, averaging versus time may be more useful.

Figure 9 on the next page schematically shows the procedure applied for this analysis.

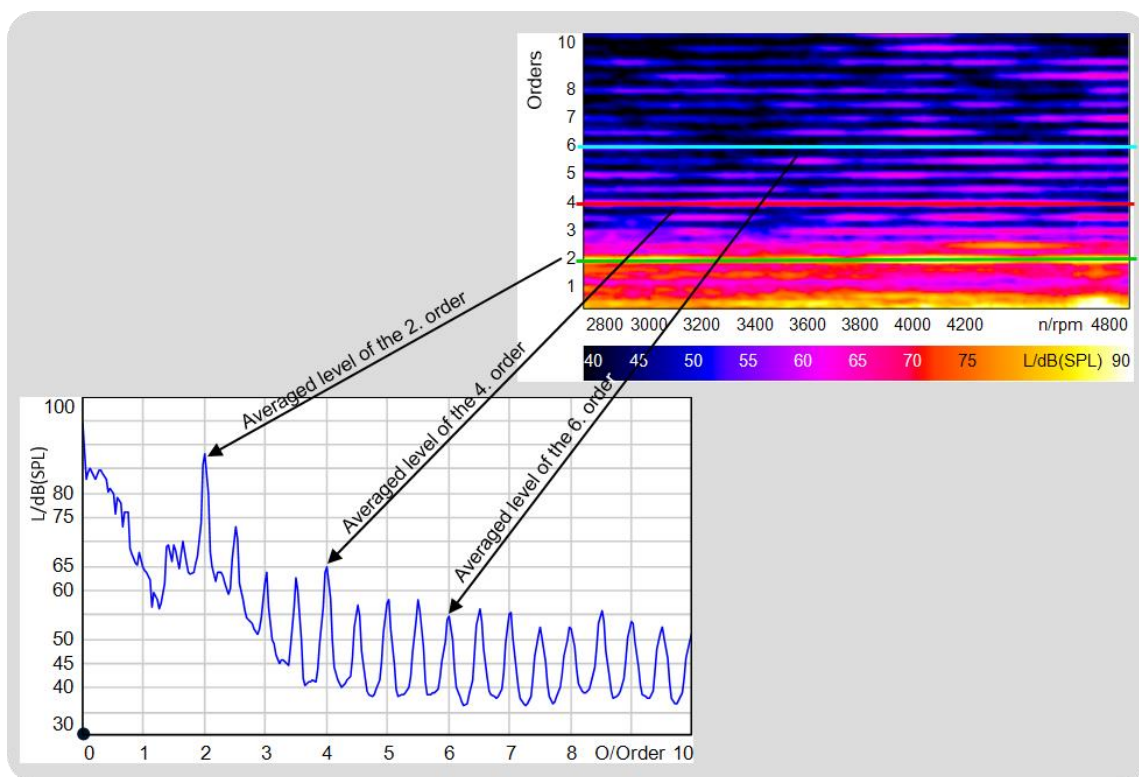


Figure 9: Averaged order analysis

Calculating an order spectrum with the *RPM-synchronous Resampling algorithm*

The **Variable DFT Size** algorithm presented above assumes that the frequency of an order remains approximately constant within the time window ΔT . This means this algorithm is only suitable for recordings where the rotational speed does not change too quickly. When an order analysis is performed on data where the rotational speed changes very rapidly, the orders “smear” when the **Variable DFT Size** algorithm is used. To avoid this smearing effect, a different calculation method called **RPM-synch. Resampling** was implemented.

In this method, a sampling rate conversion of the signal is performed first, so that the signal is no longer sampled in equidistant time intervals, but in equidistant rotation angle intervals (“resampling” of the signal). The Fourier transformation of a signal sampled like this results directly in an order spectrum⁶ and the analysis window contains the same number of signal samples at each sampling point. That means that at higher revolution speeds the signal is sampled faster accordingly, so the frequencies within the analysis window are not smeared. The sampling intervals and the DFT window width are automatically adjusted so that the desired order range and the desired order resolution defined in the Properties window are achieved.

The RPM-synchronous resampling is especially well suited for the following cases:

- Recordings with rapid RPM changes
- High resolution of orders
- Analysis of high orders

Since the level calculations of high orders in this method require very high sampling rates, the necessary computing time increases significantly if a large order range (**Spectral Range**) is selected. To reduce the processing time, the analysis should therefore be limited to the order range of interest.

Figure 10 shows the difference between the two calculation algorithms. The diagrams on the left show the results achieved with the **Variable DFT Size** algorithm, whereas the diagrams on the right show the results using **RPM-synch. Resampling**. For the two upper diagrams, an order resolution of **0.1** was selected, for the two lower diagrams an order resolution of **0.02** was used.

At the lower resolution (**0.1**), hardly any difference is recognizable between the two methods. However, the difference can easily be seen if a higher resolution (**0.02**) is chosen. In this case, the **RPM-synch. Resampling** method makes the individual orders appear much clearer and sharper.

⁶ By comparison, the Fourier transformation of a signal sampled equidistantly in time results in a frequency spectrum, which can be converted into an order spectrum if the RPM values are known.

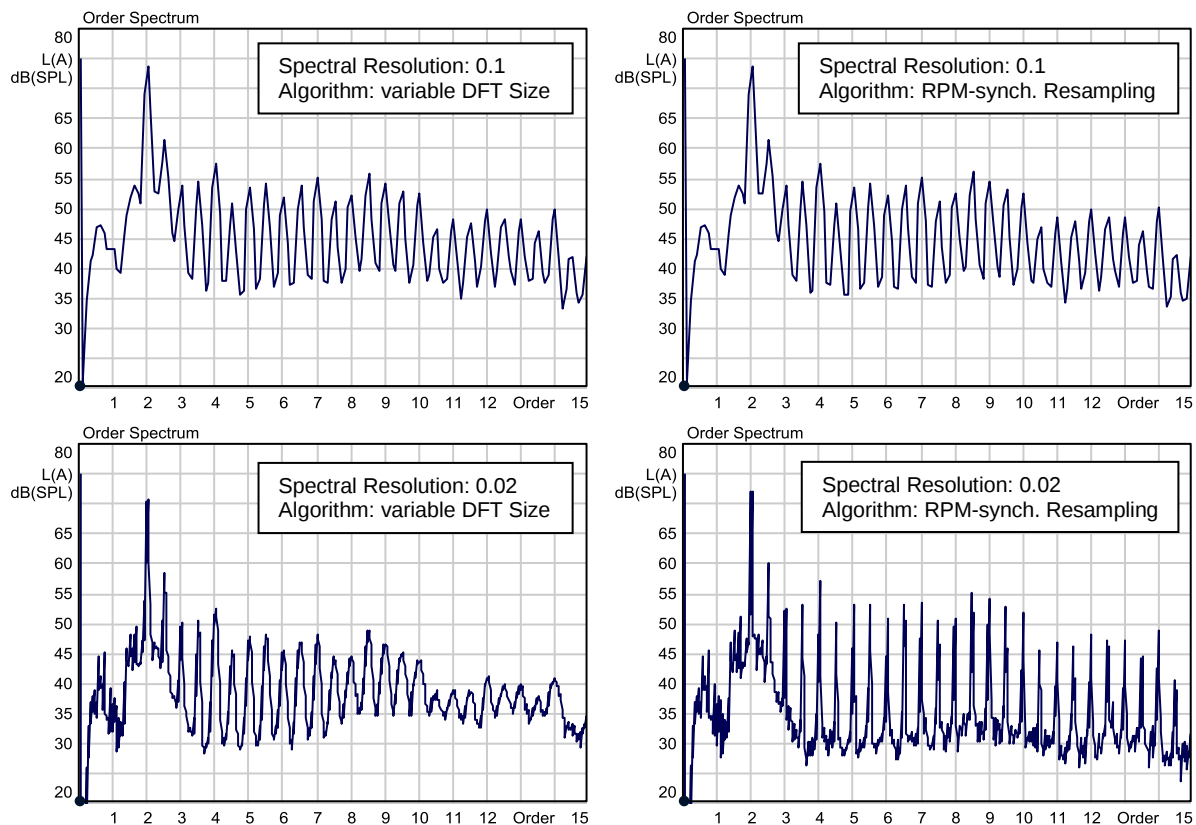


Figure 10: Comparison of the two calculation algorithms **Variable DFT Size** and **RPM-synch. Resampling** using different order resolutions

Calculating an order spectrum with the *Time Domain Averaging* algorithm

The **Time Domain Averaging** algorithm applies RPM-synchronous sampling rate conversion as well. In addition, this algorithm averages signal sections with the same phasing in the time domain versus the angle of rotation. This allows signal components that are not synchronous to the RPM orders, i.e. randomly change their phasing compared to the RPM signal, to be increasingly suppressed with increasing averaging time. The length of the signal sections is the reciprocal of the **Frequency Resolution** specified in the Properties window.

When calculating an order analysis with the **Time Domain Averaging** algorithm, it is important to know that after the time domain averaging, the order oscillations themselves are only preserved with their full level if their phasing relative to the RPM signal is stable during the averaging time.

In the averaged order analysis, time domain averaging is applied across the entire signal curve. In an order analysis versus RPM or time, averaging only takes place across the signal section corresponding to the configured step width. With a small step width (or a high order resolution), no averaging takes place at all.

Choosing the best Settings

For the proper calculation of an order analysis, it is important to choose the appropriate settings in the Properties window. However, there are no general rules as to which settings work best for an analysis. They must be selected according to the specific requirements for the desired analysis results (e.g. good time resolution or good order resolution). Furthermore, the increased computing time requirements of the algorithms **RPM-synch. Resampling** and **Time Domain Averaging** must be considered. The following table provides an overview of possible requirements and the corresponding recommended settings:

Requirement	Settings
High time resolution	• Low order resolution in the field Spectral Resolution • Small sampling intervals in the field Step Size
High order resolution	• High order resolution in the field Spectral Resolution • RPM-synch. Resampling or Time Domain Averaging algorithm for recordings with rapid RPM changes
Low computing time	• Variable DFT Size algorithm • Small order range in the field Spectral Range
Smoothed curves	• Setting Fast in the field Time Weighting

Table 1: Analysis requirements and corresponding recommended settings

Note

For calculating the analyses presented in this Application Note by means of a Pool Project, you need the following ArtemiS SUITE modules: **ASM 00** ArtemiS SUITE Basic Framework (code 5000), **ASM 01** ArtemiS SUITE Basic Analysis Module (code 5001), and **ASM 13** ArtemiS SUITE Signature Analysis Module (code 5013).

If you want to calculate the analyses by means of an Automation Project or a Standardized Test Project, you may need other modules. Your HEAD acoustics representative will gladly provide you with further information.

Do you have any questions or comments?
Please write to imke.hauswirth@head-acoustics.de.
We look forward to receiving your feedback!