Manual

Measuring the relative timing jitter of a dual-comb

Sandro Camenzind (casandro@phys.ethz.ch)

Contents

1 Purpose of this manual

1.1 What this manual does provide

This manual gives a practical guideline to implement relative timing noise measurements (particularly also the corresponding power spectral density) for a dual-comb system as discussed for example in the following publications:

- S. L. Camenzind, D. Koenen, B. Willenberg, J. Pupeikis, C. R. Phillips, and U. Keller, "Timing jitter characterization of free-running dual-comb laser with sub-attosecond resolution using optical heterodyne detection," Opt. Express **30**, 5075-5094 (2022)
- C. R. Phillips, B. Willenberg, A. Nussbaum-Lapping, F. Callegari, S. L. Camenzind, J. Pupeikis, and U. Keller, "Coherently averaged dual-comb spectroscopy with a low-noise and high-power free-running gigahertz dual-comb laser," Opt. Express **31**, 7103-7119 (2023).

For this purpose, we provide a step-by-step introduction to build the measurement setup, record the signal and analyze the traces to retrieve the relative timing noise power spectral density of a dual-comb system.

1.2 What this manual does not provide

The aforementioned publications provide the theory for the herein discussed measurement, as well as the motivation for measuring the relative timing noise of a dual-comb laser which is a crucial parameter for most dual-comb applications. For the sake of brevity, we thus refer to those manuscripts if the reader would like to complement this manual with a more detailed discussions of the measurement principle.

2 Required components

2.1 List of required components

The following components are **required** for the measurement:

Item	Quantity	Notes
Photodetector	3	Bandwidth $\geq f_{\text{rep}}/2$
cw-laser	2	Need to emit at different wavelengths
Oscilloscope	1	With a sampling rate $\geq 1/f_{\text{rep}}$ and ≥ 3 channels
Beam splitter	6	Can be in fiber, free-space or on-chip
PC for data analysis	1	Ensure that sufficient RAM is available to perform an FFT of the measured traces

2.2 List of optional components

The following components are **beneficial** for the measurement, particularly for obtaining a low measurement noise floor:

Item	Quantity	Notes
Electronic low-pass filter	3	Cut-off frequency $< f_{\text{rep}}$ to suppress the repetition rate peak
Electronic amplifiers	3	Bandwidth $\geq f_{\text{rep}}/2$, ideally low-noise amplifiers
High-finesse etalon	1	Ideally with a free-spectral range that matches the separation between the cw-laser wavelengths

Electronic low-pass filter: With the low-pass filter we can suppress the repetition rate which helps to better exploit the dynamic range of the oscilloscope.

Electronic amplifier: The amplifier increases the signal strength which is beneficial to reduce the relative noise contribution from e.g. the oscilloscope.

High-finesse etalon: For the traces where we detect the beating of the cw-lasers with the combs, only the comb-lines close to the cw lasers are contributing to the beat-notes as will be discussed in more detail below. By sending the combs through an etalon prior to detection (they can e.g. cross in the etalon), we can suppress a lot of the comb's optical spectrum which is not contributing to the beat-note. This allows to increase the optical power of the relevant portions of the optical spectrum (i.e. those close to the cw-laser wavelengths) without saturating the photodetector. This is beneficial for a low measurement noise floor.

2.3 Selection of the cw-lasers

Wavelength: The cw-lasers should emit at different wavelengths, and ideally at least one of them should be tuneable so that its wavelength can be positioned symmetrically to the other cw-laser at opposite sides of the comb's optical spectrum.

Having the two cw-lasers on the same side of the comb's optical spectrum would still work, but leads to a higher noise floor in the resulting power spectral density, which is inversely proportional to ΔN^2, where ΔN are the number of comb-lines between the two cw lasers.

Single-mode: The cw-lasers should emit single-mode beams to allow for a good overlap with each other and the combs.

Stability: Frequency-fluctuations of the cw-lasers are largely suppressed as part of the analysis routine, but the cw-lasers should still be sufficiently stable to allow for an unambiguous tracking of the beat-notes. The specific stability-requirement depend on the desired trace length, stability of the combs and their repetition rate frequency.

3 Building the setup

In this section we provide a step-by-step guide to build the measurement setup. In the following illustrations of this process, the setup is based on free-space components, but it could also be implemented with fibers or waveguides.

3.1 Install the dual-comb laser

We start from the dual-comb laser to be analyzed. It has two output beam which are denoted as comb 1 (red) and comb 2 (blue):

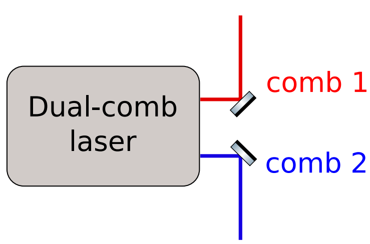

Figure 1: Dual-comb laser with two output beams: comb 1 (red) and comb 2 (blue).

3.2 Channel 1: Dual-comb interferograms

To get the dual-comb interferograms (on channel 1) we combine the two combs using a beam splitter (here depicted with a beam splitter cube) and send them on a photodetector (PD). As discussed previously in Section 2.2, we can use a combination of low-pass filters and amplifiers to improve the signal-to-noise ratio leading to a lower measurement noise floor. The optimal choice of low-pass filters and amplifiers depends on the laser characteristics, as well as the photodetector and oscilloscope used.

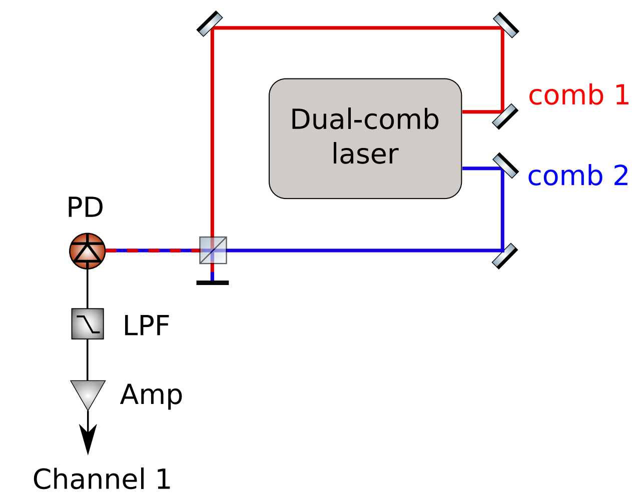

Figure 2: The two combs are combined with a beam-splitter and sent on a photodetector (PD). For optimizing the signal-to-noise ratio we use a low-pass filter (LPF) and amplfier (Amp) before recording the signal on channel 1 of the oscilloscope.

3.3 Split off part of the combs

For measuring the relative timing jitter between the combs, we also need to record the so-called beat-note signals. For that purpose, we start by splitting off part of each comb using a beam splitter:

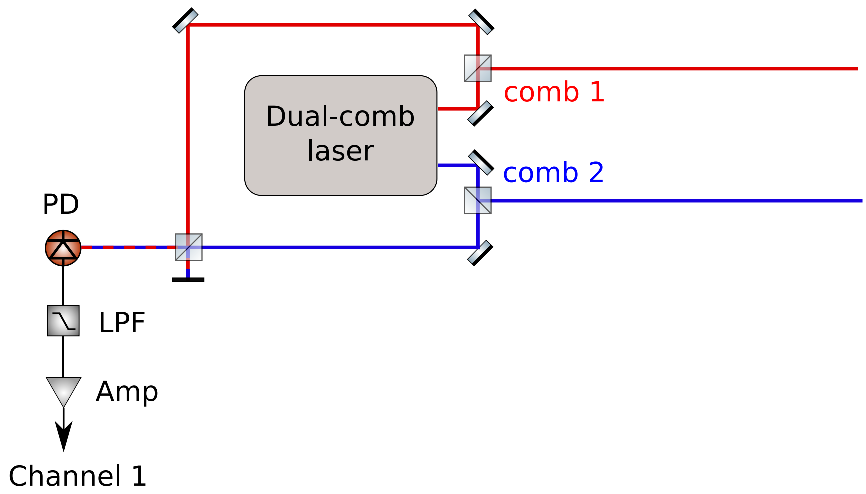

Figure 3: With a beam-splitter we split off a portion of each comb.

3.4 Prepare cw-lasers

Next, we prepare the two cw-lasers and combine their output beams with a beam splitter. This results in two spatially separated beams each containing two distinct wavelengths:

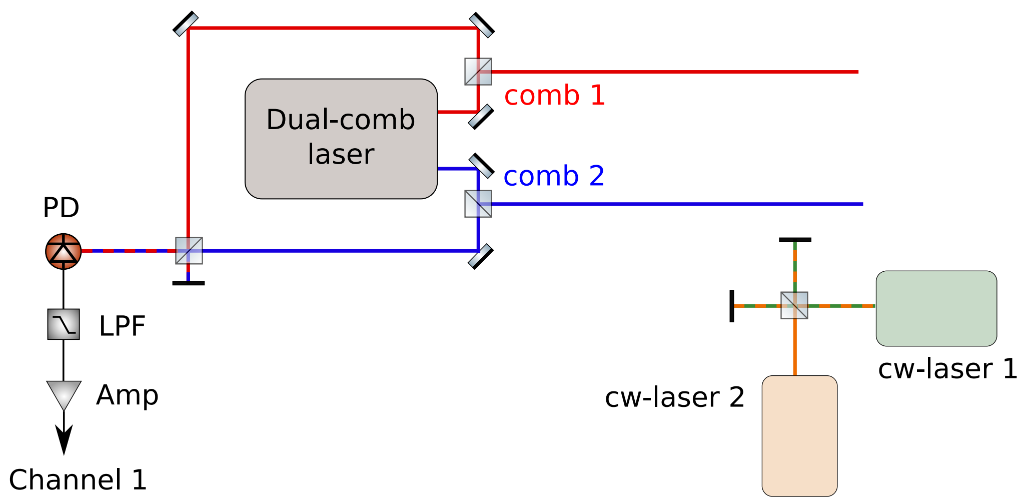

Figure 4: The output of two cw-lasers is combined with a beam-splitter.

3.5 Combine cw-lasers with the combs

To generate the beat-notes of the combs with the pair of cw-lasers, we overlap the combined cw-lasers with each of the combs. Since the measurement relies on interferometry, the signal strength will depend on the overlap-integral and the relative polarization of the beams. It is thus important to match the beam-size and polarization of the cw-lasers to the two combs.

In a free-space setup, the polarization can be controlled with waveplates, in fiber-based setup, for-example, with fiber polarization controllers or polarization-maintaining fibers.

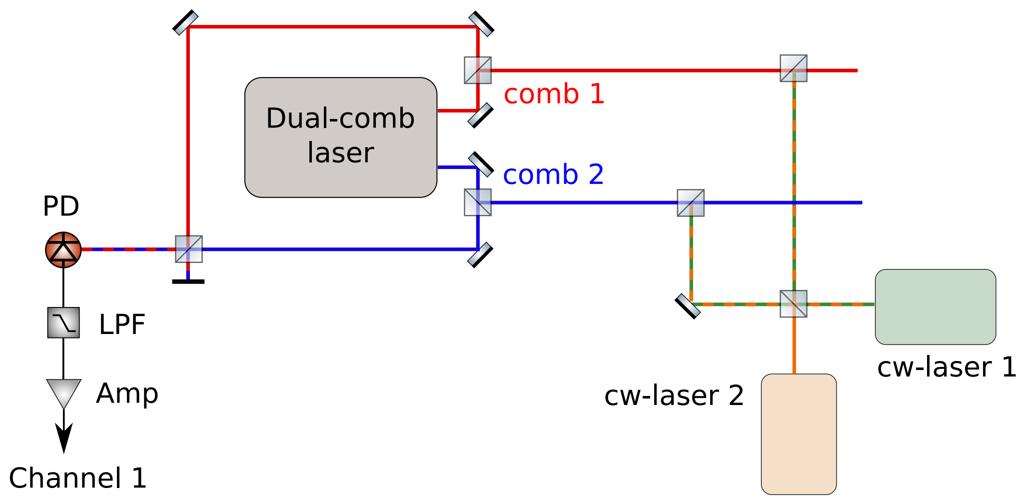

Figure 5: The combined cw-laser output is overlapped with each of the combs to generate the beat-note signals.

3.6 Channel 2 and 3: Beat-note signals

The resulting combinations of comb i (i $\in \{1, 2\}$) and the two cw-lasers are directed onto corresponding photodetectors. Again, to improve the signal-to-noise ratio, we can use low-pass filters and amplifiers before recording the trace with the oscilloscope on channel 2 and channel 3.

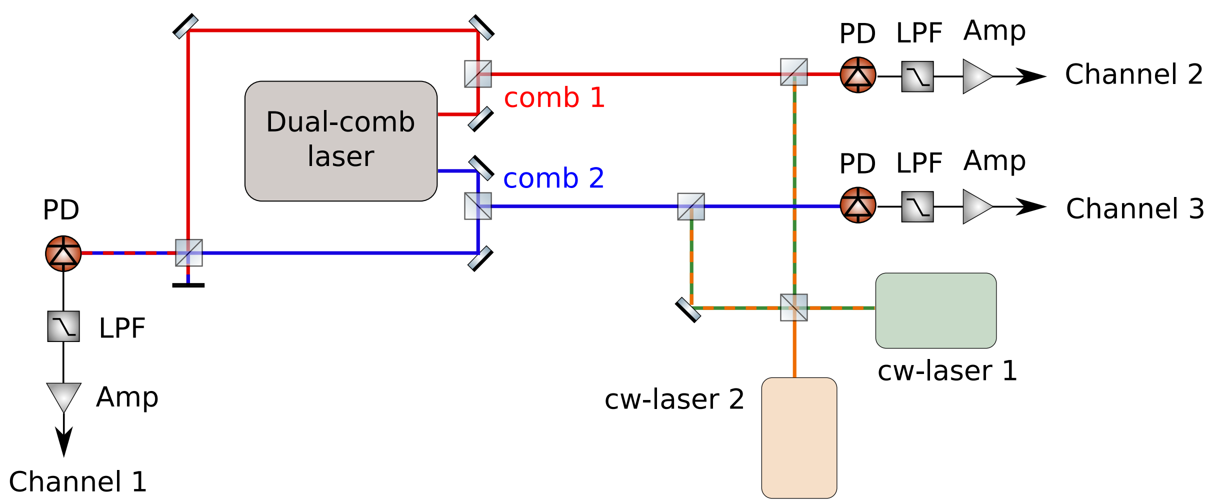

Figure 6: The beat-note signals are detected with a corresponding photodetector (PD). The resulting signal is recorded on channel 2 and channel 3 of the oscilloscope. A combination of low-pass filters (LPFs) and amplifiers (Amps) can be used for optimizing the signal-to-noise ratio.

4 Measurement signal

In this section we show the typical structure of the recorded traces. This can be helpful to know what kind of signals to expect when recording the signal.

4.1 Channel 1: Dual-comb interferograms

On channel 1 we record the dual-comb interferograms (Fig. 7(a)), which in the frequency-domain result in the dual-comb spectrum (Fig. 7(b)):

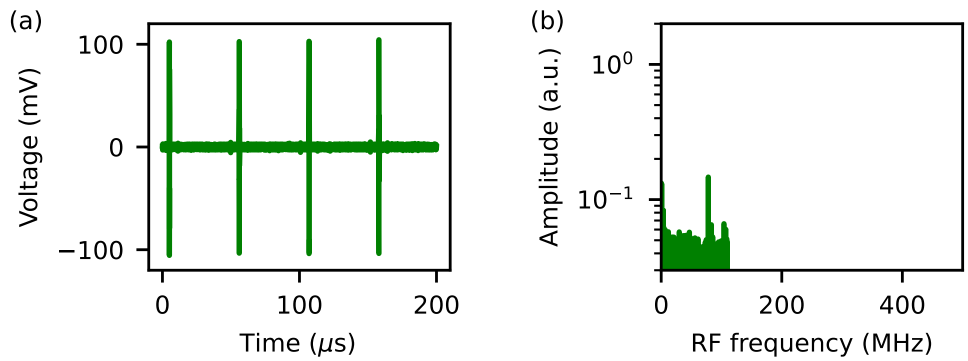

Figure 7: (a) Dual-comb interferograms in the time-domain. (b) Dual-comb spectrum in the frequency domain.

4.2 Channel 2: Comb 1 + cw-lasers

On channel 2, we have the beating of comb 1 with the two cw-lasers. In the time-domain this results in a combination of sinusoidal signals at the beat-note frequencies between the cw-laser and the comb lines (Fig. 8(a)). This is visible in the frequency-domain where the beat-notes appear as two peaks below $f_{\mathrm{rep}}/2$ ($\approx$ 500 MHz for the dual-comb laser characterized in this measurement), one for each cw-laser beating with the closest comb line (Fig. 8(b)).

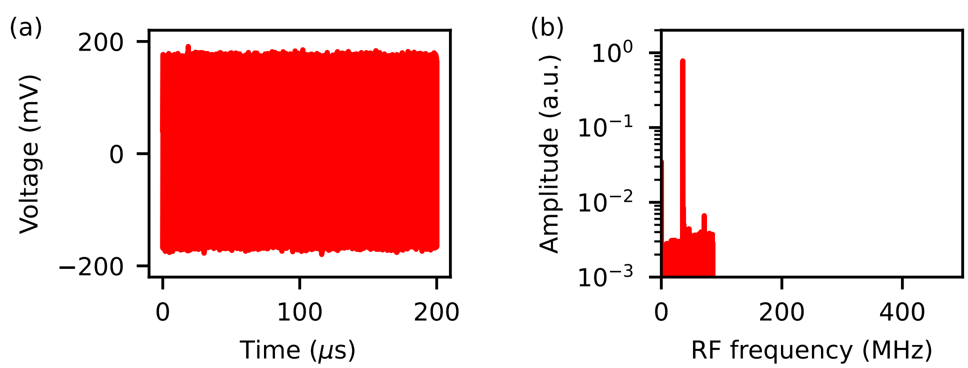

Figure 8: (a) Time-domain representation of the beat-note signals between comb 1 and the cw-lasers. (b) In the frequency-domain, the beat-notes appear as distinct peaks.

Note: We found that a signal-to-noise ratio of $\geq$ 20 dB for the individual beat-notes is desirable to allow for accurate tracking of their phase.

4.3 Channel 3: Comb 2 + cw-lasers

In Fig. 9 we have the beat-note of the cw-lasers with comb 2. Compared to the measurement shown in Fig. 8 for comb 1, the beat-notes are now at different frequencies since the two combs may have a different pulse repetition frequency and carrier-envelope offset frequency leading to shifted optical comb-lines.

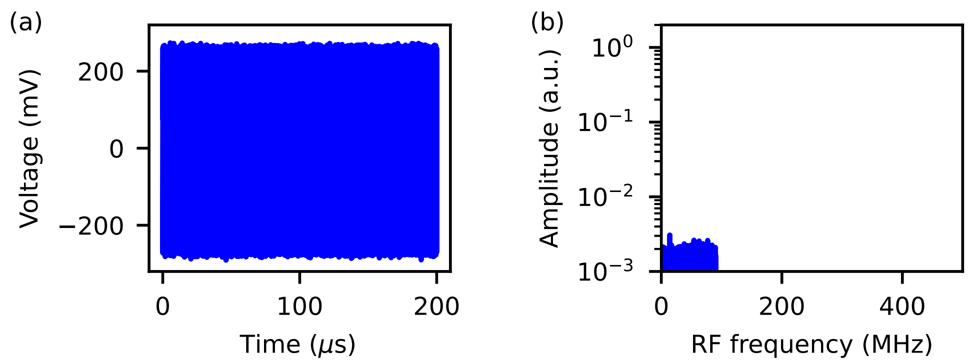

Figure 9: (a) Time-domain representation of the beat-note signals between comb 2 and the cw-lasers. (b) In the frequency-domain, the beat-notes appear as distinct peaks.

5 Data analysis

5.1 Repository structure

The analysis code is available on GitHub (https://github.com/ca-sandro/Dual-Comb-RTJ.git). The repository contains:

Type	Name	Notes
Python file	analyze_BN.py	main script to process the signals and generate the relative timing noise power spectral density
Markdown	README.md	Fundamental instructions to analyze the beat-note data
Folder	DC_data	Contains specific functions that are relevant for the beat-note processing
Folder	basics	Contains more general functions for the data analysis
Folder	example_data	Contains example data (stored as numpy arrays) for the three channels.
PDF	manual.pdf	This is the manual you are reading

5.2 Prepare the loading of the data

5.2.1 Name of the traces

The data should be stored as numpy arrays, one for each channel. The files should be named according to the following system:

- **Interferograms:** 'C1' + filename_core + '.npy'

- **Beat-notes (first comb with cw-lasers):** 'C2' + filename_core + '.npy'
- **Beat-notes (second comb with cw-lasers):** 'C3' + filename_core + '.npy'

Here, filename_core can be any string. For example if filename_core = '_trace', the file-names would be:

- **Interferograms:** 'C1_trace.npy'
- **Beat-notes (first comb with cw-lasers):** 'C2_trace.npy'
- **Beat-notes (second comb with cw-lasers):** 'C3_trace.npy'

5.2.2 Set the path in analyze_BN.py

To tell the code where the data is stored and how the traces are named, one has to:

- Store the path that points to the folder containing the traces in the variable **dir_data**
- Store the filename_core as described previously in Section 5.2.1 in the variable **filename_core**

5.3 Provide the processing parameters

For the processing, a few parameters have to be stored in the corresponding variable within the main script analyze_BN.py:

- **f_rep_approx:** Repetition rate of the dual-comb laser (approximately)
- **D_frep_approx:** Repetition rate difference of the dual-comb laser (approximately)
- **dt:** Sampling time-step of oscilloscope
- **nt:** Number of data-points in a single trace

5.4 Select if the traces have to be downsampled

For the traces recorded on channel 2 and channel 3, we are only interested in the individual beat-notes, which typically occupy a narrow region in the frequency domain as visible in Figs 8(b) and 9(b). To reduce the processing time, it is thus sensible to extract and downsample those beat-note signals.

This downsampling has to be performed only once for each trace, since we store the downsampled data so that for subsequent executions of the code the downsampled data can be loaded. The downsampling-routine can be enabled/disabled by setting the variable **downsample_BN** to True/False.

5.4.1 For downsampe_BN = True

If the beat-note signals have to be downsampled, the code displays the beat-notes in the frequency domain (similar to the data shown in Figs. 8(b) and 9(b)) as follows:

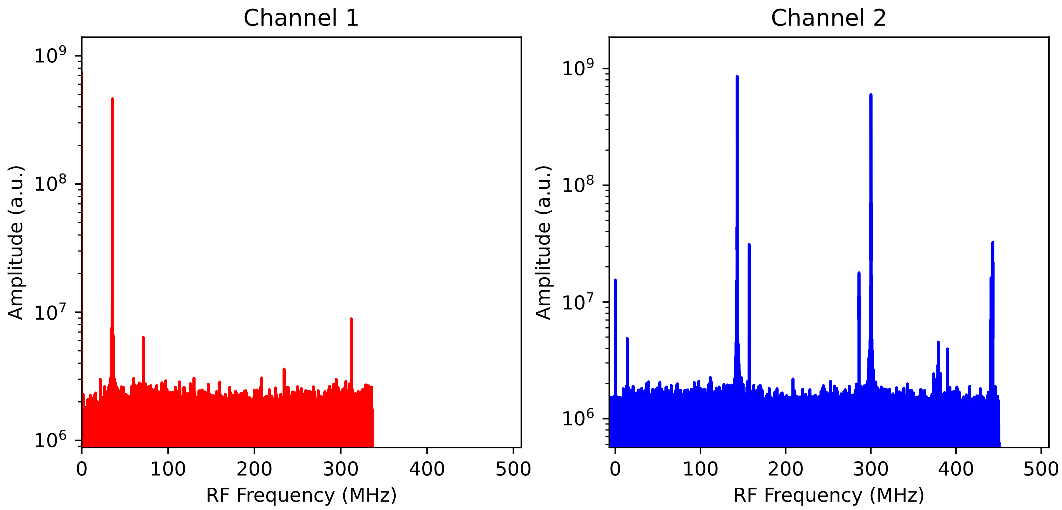

Figure 10: Frequency-domain representation of the four beat-note signals. In red are the beat-notes from channel 1 and in blue from channel 2.

The user has to estimate the center-frequency and width of the beat-notes and enter them in the corresponding field in the terminal. The code then proceeds with the downsampling and after completion stores the downsampled signals and supplementary information in the folder where the original traces are located. Furthermore, for each beat-note it displays the gradient of the phase (Fig. 11) as well as its frequency-domain representation (Fig. 12).

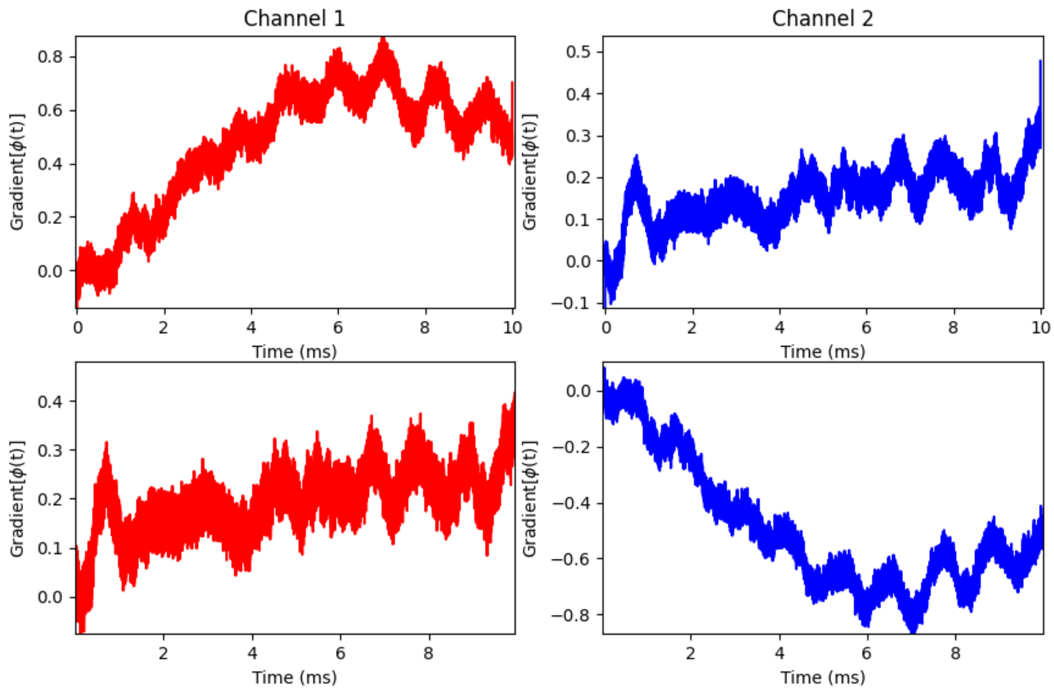

Figure 11: Gradient of the phase of the four beat-notes. In red are the beat-notes from channel 1 and in blue from channel 2.

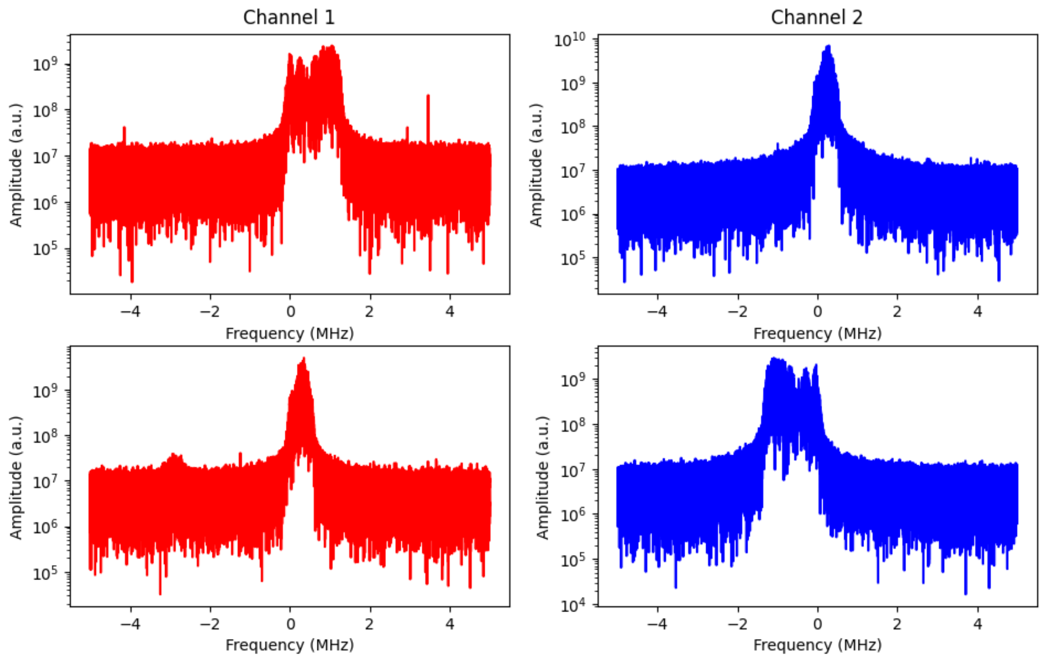

Figure 12: Frequency-domain representation of the beat-note signals. In red are the beat-notes from channel 1 and in blue from channel 2.

From Figs. 11 and 12 we can see that the beat-note on the top-left and bottom-right likely correspond to the same cw-laser (and similarly the beat-note on the top-right and bottom-left) as their phase-fluctuations are correlated (Figs. 11) and since the beat-notes have a similar width (Figs. 12).

Note: For long traces, the downsampling can take several minutes due to the Fourier-transform, which is part of the downsampling routine.

5.4.2 For downsamlpe_BN = False

If the downsampling was already done previously, downsampe_BN can be set to False. In this case, the downsampled data is loaded which drastically reduces the processing time.

5.5 Result of the analysis

As a result of the analysis, the code provides the calculated frequency-difference between the cw-lasers, as well as ΔN, the integer number of comb-lines between the cw-lasers.

Furthermore, it displays the relative timing jitter power spectral density and integrated relative timing jitter for the dual-comb as shown in Fig. 13:

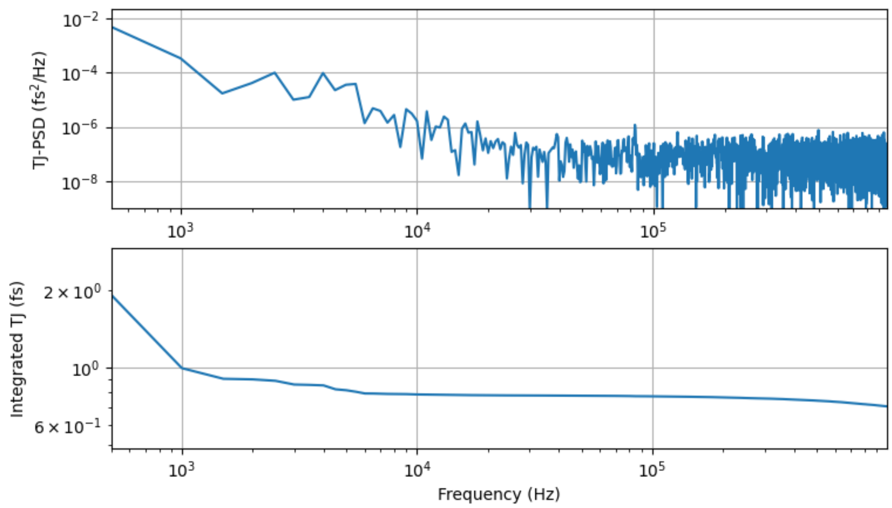

Figure 13: Result of the analysis: Relative timing jitter power spectral density and integrated relative timing jitter for the dual-comb.

5.6 ERROR - integer number of comb lines not found

If you receive this error message, it means that the code was not able to unambiguously determine the variable ΔN, i.e. the integer number of comb-lines between the two cw-lasers.

The most likely reason for this is that through the calculation an offset $n \cdot f_{\text{rep}}, n \in \mathbb{Z}$ was introduced. It is possible to correct for this by subtracting this offset, but this requires accurate knowledge of f_{rep} to prevent errors in the estimated value of ΔN. Alternative options which do not require this information are provided by this code via the following settings:

1. **flip_BN_order = True**: This setting flips the order of channel 2 and channel 3, which can already be sufficient to resolve the aformentioned problem.
2. **select_higher_BN = True**: This allows the user to select higher-frequency beat-notes above $f_{\text{rep}}/2$. In this case, it is necessary to also set downsample_BN = True again since new beat-note signals are potentially selected and downsampled.

6 Explanation of the analysis

6.1 Track Δf_{rep} via the IGM arrival times

In the first step of the analysis, we use the interferograms (IGMs) recorded on channel 1 to coarsely track Δf_{rep} via the IGM arrival times. For that purpose, the code initially does a pre-processing of the first few IGMs to extract their duration and carrier-frequency which helps to determine the IGM arrival times more accurately for the rest of the trace:

6.1.1 Pre-processing of the interferograms:

1. We load the first few IGMs for which we want to perform the pre-processing
2. Next, we calculate the envelope of the IGMs via the Hilbert transform to allow for a more accurate peak-finding without being susceptible to the oscillations of the IGM signal.

3. From the envelope we extract the IGM duration in the time-domain and the approximate arrival time of the IGMs using scipy.signal.find_peaks(). Furthermore, we investigate the IGM signal in the frequency domain to determine its carrier-frequency.
4. We then leverage the estimated carrier-frequency to apply a band-pass filter around the IGM signals in the frequency domain, then calculate their envelope using the Hilbert transform and finally utilize the estimated duration of the signal to zoom-in around the IGM-envelopes and refine their estimated arrival time using a second-order moment integral. The IGM arrival time provides an accurate (but coarsely sampled) measurement of Δf_{rep} for the duration of the first few IGMs analyzed in the pre-processing.

This information is then used for finding the IGM arrival time also for the rest of the trace:

6.1.2 Processing of the interferograms

1. We start from the previous IGM peak and want to predict the arrival time of the following IGM signal. For that purpose, we approximate its position from the previous IGM peak and Δf_{rep}.
2. After applying the digital band-pass filter and computing the envelope using the Hilbert-transform, we use scipy.signal.find_peaks() to get a better estimate of the IGM position.
3. Finally, we zoom in around the IGM and use a second-order moment integral to further refine the IGM arrival time (similar to step 4 in the previous Section 6.1.1).

This is repeated until we reach the end of the trace, which results in an accurate measurement of the IGM arrival times and thus Δf_{rep} for the entire duration of the trace.

6.2 Downsample the beat-notes

To downsample the beat-notes in channel 2 and channel 3 (as discussed in Section 5.4), the code

1. First performs a Fast Fourier Transform (FFT) of the traces.
2. Then selects the four beat-notes according the the corresponding center-frequency and bandwidth provided by the user.
3. Finally performs and inverse FFT (iFFT) to receive the downsampled signal in the time domain.

6.3 Calculate the product of the beat-notes

6.3.1 Remove the influence of cw-lasers

To remove the influence of the cw-laser, we take the product of beat-notes resulting from the beating of the combs with the same cw-laser. To determine which two beat-notes correspond to the same cw-laser, we:

1. Calculate the product of all possible combination of beat-notes, i.e.
   - $B^{\mathrm{comb-1}}_{\mathrm{cw-i}} \times B^{\mathrm{comb-2}}_{\mathrm{cw-j}}, \mathrm{i}, \mathrm{j} \in \{1, 2\}$
   - $B^{\mathrm{comb-1}}_{\mathrm{cw-i}} \times \overline{B^{\mathrm{comb-2}}_{\mathrm{cw-j}}}, \mathrm{i}, \mathrm{j} \in \{1, 2\}$

   where $B^{\mathrm{comb-1}}_{\mathrm{cw-i}}$ stands for the beat-note of comb 1 with cw-laser i, and $\overline{B}$ denotes the complex-conjugate of B.
2. From the product, we then extract the phase fluctuations and estimate for which two combinations of beat-notes the phase fluctuations are minimal. These combinations correspond to the two pairs of beat-notes beating with the same cw-laser since for those the (typically) dominant cw-laser fluctuations cancel.

3. Using this information, we compute the product between the corresponding beat-notes and store them for the next step. We refer to this product between the beat-notes as $B_{\mathrm{cw-i}}, \mathrm{i} \in \{1, 2\}$.

6.3.2 Remove the influence of Δf_{CEO}

To also remove the influence of Δf_{CEO}, we compute

- $B_{\mathrm{cw-1}} \times B_{\mathrm{cw-2}}$
- $B_{\mathrm{cw-1}} \times \overline{B_{\mathrm{cw-2}}}$

and, similar to the approach in the previous Section 6.3.1, we calculate the corresponding phase and check for which combination its fluctuations get minimized. For the resulting combination, we compute the respective product resulting in $B_{\mathrm{signal}}(t)$.

6.4 Compute the repetition rate difference $\Delta f_{\mathrm{rep}}(t)$

Since we know that for the frequency $f_{\mathrm{signal}}(t)$ of B_{signal} the following relation holds:

$$\Delta f_{\mathrm{rep}}(t) = \frac{f_{\mathrm{signal}}(t)}{\Delta N}, \tag{1}$$

we need to find the number of comb-lines between the two cw-lasers to extract $\Delta f_{\mathrm{rep}}(t)$. For that purpose

1. From the measured IGM arrival times and $f_{\mathrm{signal}}(t)$ we estimate how much the phase ϕ_{signal} associated with $f_{\mathrm{signal}}(t)$ evolves on average from one IGM-peak to the next. We refer to this quantity as $\Delta\phi_{\mathrm{signal,avg}}$
2. From Eq. 1 we find that

$$\Delta\phi_{\mathrm{signal,avg}} = 2\pi f_{\mathrm{signal}} \times \frac{1}{\Delta f_{\mathrm{rep}}} = 2\pi\Delta N, \tag{2}$$

   so that dividing $\Delta\phi_{\mathrm{signal,avg}}$ by 2π gives ΔN.

3. By plugging ΔN into Eq. 1 we finally obtain the repetition rate difference as a function of time $\Delta f_{\mathrm{rep}}(t)$. Compared to the coarsely sampled Δf_{rep} measured via the IGM arrival times, the $\Delta f_{\mathrm{rep}}(t)$ obtained via the beat-notes is sampled with a much higher timing resolution which allows the calculation of the associated power spectral density (PSD) up to higher noise frequencies.

6.5 Extract the timing jitter power spectral density

From $\Delta f_{\mathrm{rep}}(t)$, or more specifically from the corresponding phase $\phi_{\Delta f_{\mathrm{rep}}}(t) = \phi_{\mathrm{signal}}(t)/\Delta N$, we compute the associated one-sided phase-noise PSD via:

$$S_\phi(f) = 2 \times |\mathbb{F}(\phi_{\Delta f_{\mathrm{rep}}}(t))|^2, \tag{3}$$

where $\mathbb{F}$ indicates the Fourier transform. The factor of two is needed because we consider the **one-sided** PSD. For the numerical computation of the Fourier transform, we prepare the phase data so that there is no slope or offset at $t = 0$, and apply a suitably scaled apodization window. By multiplying the phase-noise PSD $S_\phi(f)$ with $1/(2\pi f_{\mathrm{rep}})^2$ we find the corresponding one-sided timing noise power spectral density

$$S_{\mathrm{TJ}}(f) = \frac{S_\phi(f)}{(2\pi f_{\mathrm{rep}})^2}, \tag{4}$$

which is also displayed by the code as a result of the analysis as shown in Fig. 13.