Project DIG-AC - A digital traceability chain for AC voltage and current

Final meeting – 19th and 20th May 2022 (CEM – Tres Cantos, Madrid)

WP3: Data processing and uncertainty estimation for quick integration

Task 3.2: Study of suitable algorithms
Task 3.3: Uncertainty estimation

SFDR algorithm validation and uncertainty estimation

IPQ – Vitor Cabral

A3.2.3 **M12**	**IPQ, GUM, CEM and CMI will select and validate the most suitable algorithms (up to 2) for calculation of parameters required for processing the data defined in A2.1.2 and A2.1.4, and those measured in A4.1.1 - A4.1.3. The selection will be based on robustness, data processing time and accuracy of algorithms. This activity will provide input to A3.3.2, A3.4.1, A3.4.2 and A6.1.8.**	**IPQ, GUM, CEM CMI**

Algorithm

SFDR – Spurious Free Dynamic Range

Available in the QWTB - Software Toolbox for Sampling Measurement (https://qwtb.github.io/qwtb/

Calculates de Spurious Free Dynamic Range of a sampled signal, in dB:

$$SFDR = 20\log(A_{spurios}/A_0)$$

Based on FFT method by calling the **SF-WFFT algorithm** (also available in the QWTB) with **Blackman windowing** (by default) to calculate the spectrum of a sampled sinewave signal.

Validation tests

- **Algorithm error** was calculated as the difference between the *SFDR* value calculated by the algorithm and the theoretical value calculated to the simulated input signal.

  For coherently and noncoherently sampling

  Repeated runs ⟶ standard deviation (only for coherently sampling)

- **Uncertainty was added to the sampled values** and the effect in the algorithm output value and the related uncertainty value estimated by the **qwtbvar.m** function (using Monte Carlo method) were observed.

  These results were compared with the SFDR algorithm error and its standard deviation obtained in previously tests.

Simulated signal

Signal quantities:

- main signal component amplitude: 1 V
- amplitude offset: 0 V
- number of signal components: 2 (main and spurious)
- signal components phase: 0 rad
- frequency of fundamental, f_0: 100 Hz to 1 kHz
- spurious component frequencies, (x f_0): 0.5, 1.1, 1.5, 2.5, 3.5, 4.5
- *SFDR* values: -40 dB, -80 dB, -120 dB and -140 dB

Acquisition and calculation settings

Acquisition quantities:

- Sampling frequency: 10 kHz => 2.2 x highest component present in signal (f_0= 1 kHz and $f_{spurious}$= 4.5xf_0)
- Record length: 10 kSa (1 second) => nbr of sampled periods: 100 (@ f_0= 100 Hz) to 1000 (@ f_0= 100 Hz)
- ADC bit resolution: 28 bits (input of qwtb.m)

Calculation settings (used by qwtb.m):

- CS.unc = 'mcm'
- CS.mcm.repeats = 1000

RESULTS

Algorithm error dependence on frequency, spurious component and *SFDR* value

SFRD algorithm was run with input sampled data and for each one of the following signal parameter values combinations:

Frequency of the fundamental (f_0): 100 Hz to 1 kHz (with 100 Hz steps)

Spurious frequency (f_s): 0,5, 1.1, 1.5, 2.5, 3.5, 4.5 (x f_0)

SFRD: -40 dB, -80 dB, -120 dB and -140 dB

Results: the calculated relative error of the *SFDR* values obtained for combinations of the input signal parameters were ≤ **6×10^{-9}**

(highest value obtained to f_0 = 1 kHz, f_s = 0.5xf_0, *SFDR* = -140 dB)

RESULTS

Algorithm error dependence on noise value

Random noise was added into simulated signals of:

f_0 = 100 Hz and 1 kHz, with the fixed values of *SFDR* = -80 dB and f_s = 1.5 x f_0

Noise values amplitude related to the sampled signal values (:
$1x10^{-6}$, $1x10^{-5}$ and $1x10^{-4}$ V

The algorithm was run 1000 times for each noise value amplitude

RESULTS

Algorithm error dependence on noise value

Noise amplitude / V	f_0	*SFDR* relative error			
		Mean value	Standard deviation	Maximum	Minimum
$1x10^{-6}$	100 Hz	$-3x10^{-7}$	$2x10^{-5}$	$7x10^{-5}$	$-7x10^{-5}$
$1x10^{-5}$		$-3x10^{-6}$	$2x10^{-4}$	$6x10^{-4}$	$-6x10^{-4}$
$1x10^{-4}$		$2x10^{-5}$	$2x10^{-3}$	$7x10^{-3}$	$-7x10^{-3}$
$1x10^{-6}$	1 kHz	$-4x10^{-7}$	$2x10^{-5}$	$7x10^{-5}$	$-6x10^{-5}$
$1x10^{-5}$		$7x10^{-6}$	$2x10^{-4}$	$6x10^{-4}$	$-6x10^{-4}$
$1x10^{-4}$		$4x10^{-5}$	$2x10^{-3}$	$7x10^{-3}$	$-6x10^{-3}$

Standard deviation values are of one order of magnitude greater than the noise value

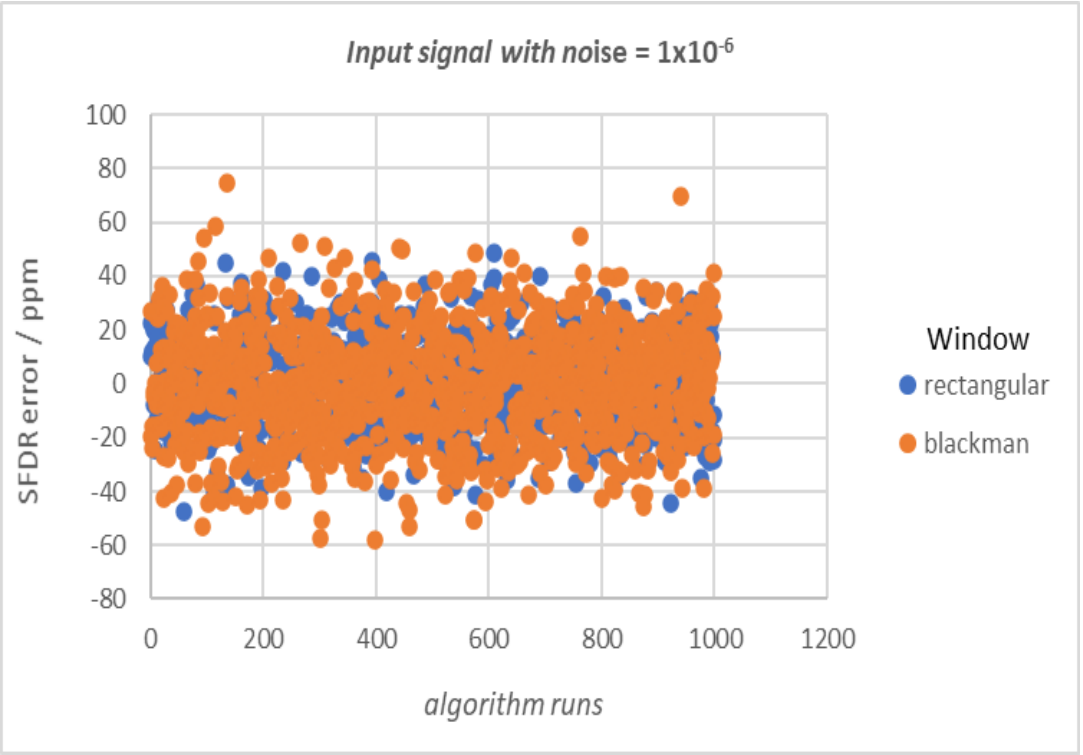
RESULTS

The algorithm was also run with a **rectangular window** instead of the default **blackman window**

For rectangular window:

standard deviation are between 21 % and 27 % lower

The use of windows in coherent sampling will increase the influence of noise in the standard deviation of the FFT results



SDFR error of the algorithm output to a simulated signal of 1 kHz, SFDR = -80 dB, spurious component at 1.5x f_0 and sampled values with $1x10^{-6}$ of random noise.

RESULTS

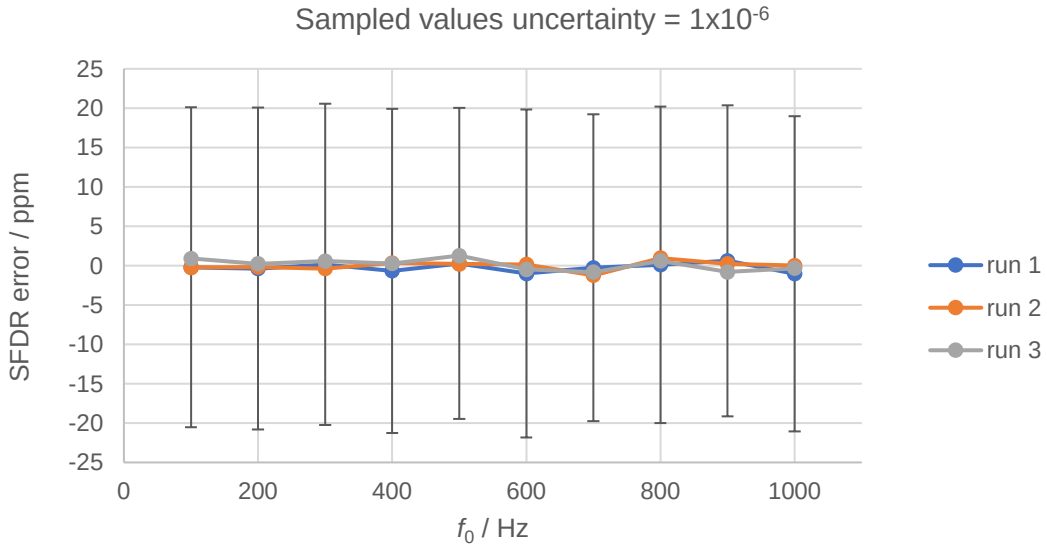
Algorithm error and uncertainty dependence on uncertainty in each sampled value

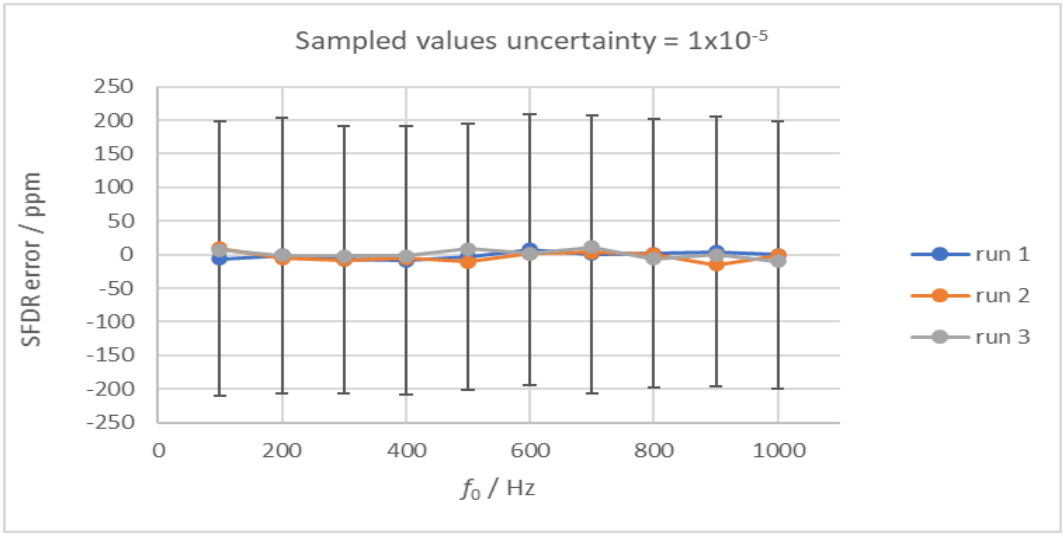
For each sampled value of the simulated signal (with *SFDR* = -80 dB and $f_s = 1.5xf_0$) was introduced an uncertainty with values of $1x10^{-6}$, $1x10^{-5}$ and $1x10^{-4}$ V

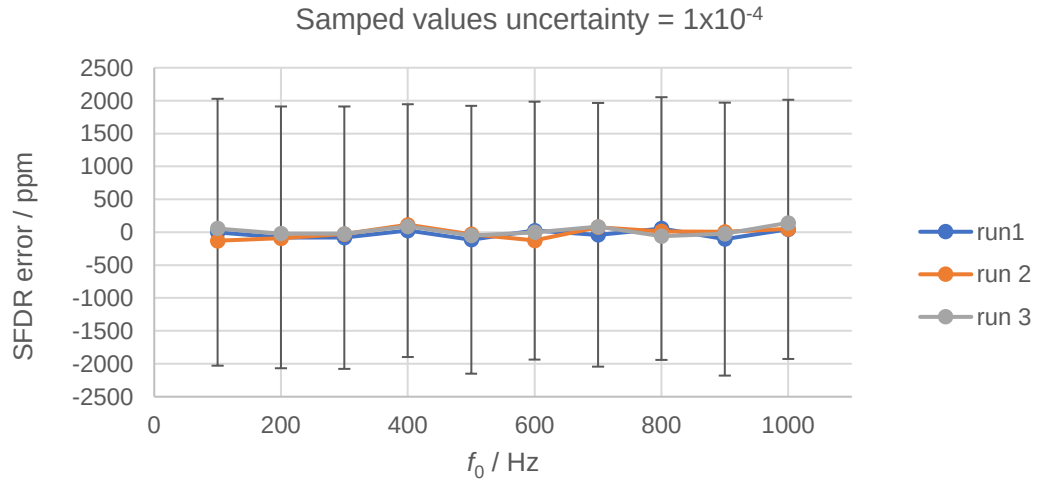
For each one of these uncertainty values, the algorithm was run 3 times

The uncertainty was calculated by the qwtb.m to the *SFDR* estimation

RESULTS







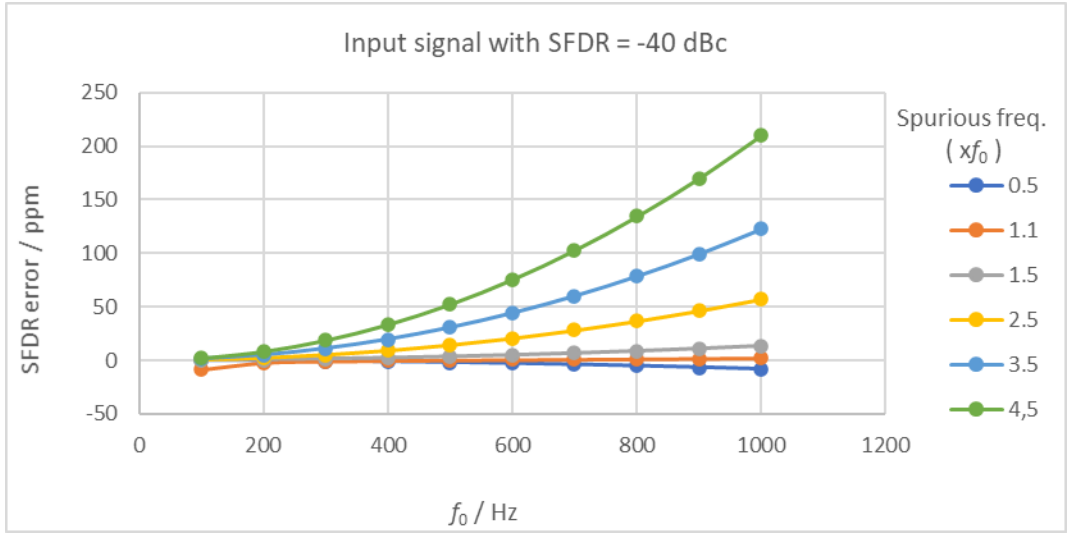
The error bars represent the uncertainty calculated by the qwtb.m to the SFDR estimation.

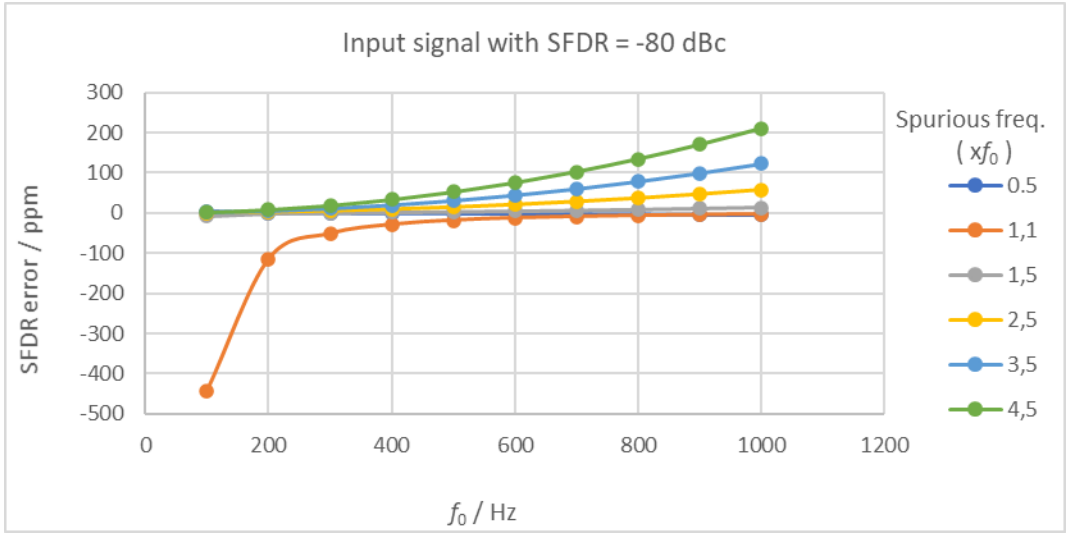
Uncertainty values are of the same order of magnitude of the standard deviation values obtained before running 1000 times the algorithm with noise added to each sample value

RESULTS

Algorithm error for noncoherent sampling

To see the performance of the algorithm with noncoherent sampling, a deviation of 10 ppm was introduced in the fundamental frequency to the simulated test signal

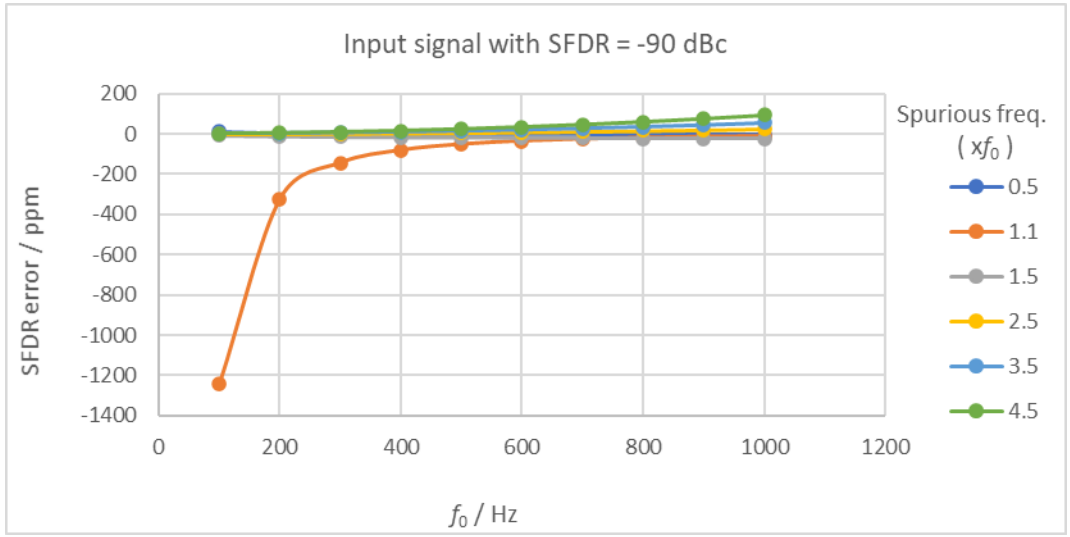




RESULTS

Algorithm error for noncoherent sampling

To see the performance of the algorithm with noncoherent sampling, a deviation of 10 ppm was introduced in the fundamental frequency to the simulated test signal



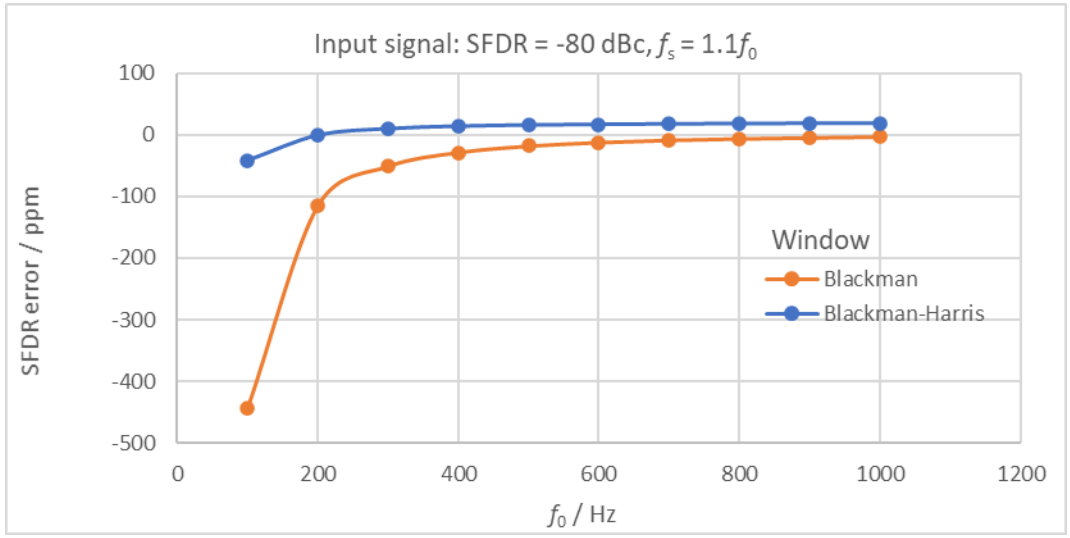
SDFR relative error depends on the input quantities of the signal: *SFDR*, fundamental frequency and non-harmonic component.

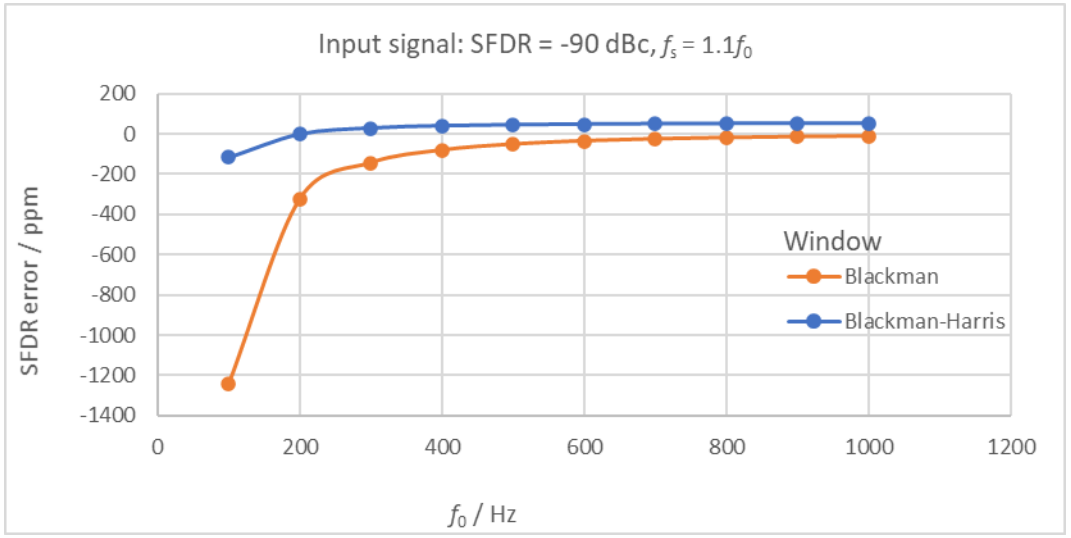
For the range of signal quantities tested, without considering the results for the spurious frequency f_s = 1.1xf_0, the largest error founded was 210 ppm to a signal with SFDR = -40 dB and with a spurious frequency equal to 4.5 times the fundamental.

RESULTS

Algorithm error for noncoherent sampling

A **Blackman-Harris window** with stronger leakage reduction than Blackman window was experimented to the signals with $f_s = 1.1xf_0$





For low frequencies the algorithm error is strongly reduced. For higher frequencies (>500 Hz), the absolute error increases

RESULTS

Algorithm error for noncoherent sampling

It was also made some tests for *SFDR* values of **-100 dB** and **-120 dB**:

For frequencies above 300 Hz to *SFDR* =-100 dB and

For all the range from 100 Hz to 1 kHz to *SFDR* = -120 dB

The errors obtained are too high to be considered as meaningful => **Algorithm is not working!**

CONCLUSIONS

SFDR algorithm presents **no relevant systematic error for a coherent sampling** of sine wave signal in the frequency range tested:

(100 Hz to 1 kHz, with non-harmonic components from 0.5 to 4.5 of the main frequency and to *SFDR* values from -40 dB to 140 dB)

The algorithm output shows dependence on the random noise value present in the sampled signal:

- with standard deviation values of **one order of magnitude greater than the random noise**.
- the **standard deviation of the result cover by two order of magnitude the relative error of the *SFDR*** estimation.

The uncertainty added to each sampled value is processed by the qwtb.m function and generate an uncertainty estimation which **is in accordance** with the algorithm dependence on random noise observed.

CONCLUSIONS

The application of SFDR algorithm to noncoherently sampled signal:

- Generate results with significant systematic error (≈ 0.12%, maximum observed) which depends on the signal parameter values.

- The window used has a strong influence in the error obtained.

- The algorithm is not working to ***SFDR*** values of **-100 dB** and **-120 dB**

Thank you very much for your attention