SUCHAI Physics Payload

Technical specifications for the onboard physics experiment

Change log

Version	Date	Author	Changes
1.0	05/07/16	José Ogalde	Initial documentation.
2.0	07/22/16	José Ogalde	Filtering and range of operation included.
2.1	08/16/16	José Ogalde	Fixing simulation figures.
3.0	11/04/16	José Ogalde	Add modification in waveforms and operations.
3.1	18/03/17	José Ogalde	Add density functions plots.

Table of contents

Physics Experiment Overview

Given that electronics are the main resource for automation, control and data acquisition for complex systems, is of paramount importance to characterize their behaviour in hostiles environments such as space, in particular when considering environmental strains and attacks on normal behaviour of electronic components by the interaction of its parts with its surroundings.

This document presents an experiment that constitutes one of the payloads in the SUCHAI cubesat. The experiment main goal is to study the hostility of space environment on electronics due to several effects such as extreme low-high temperature cycles and EM radiation. This can be accomplished by measuring the statistical properties of the fluctuations of energy flux in a dissipative system -such as a simple RC circuit- when driven with a stochastic input. Previous work by Falcón (2009) [1] gives a reference for the expected results for outer space. Hostility of the environment could be inferred comparing space with laboratory statistics.

Because of the simplicity in their constructions, cubesats present as excellent candidates to perform this kind of studies for electronics. For this purpose, the SUCHAI cubesat has an RC circuit driven with a pseudo-random voltage as one its payloads.

Experimental Setup

The circuit is a SISO system driven with a known pseudo-random voltage $v_{in}(t)$, that has a characteristic frequency f_c, and the output voltage $v_{out}(t)$ is then measured. The purpose of this experiment is to measure the output voltage $v_{out}(t)$ for different values of f_c, which when multiplied with $v_{in}(t)$ gives a value proportional to the injected power to the system.

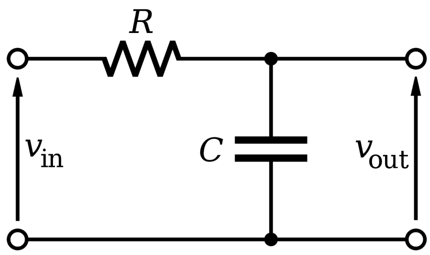


Figure 1

A PIC microcontroller controls the input voltage through an DA converter (DAC). The DAC receives the pseudo-random values generated by the microcontroller and outputs the corresponding analog values every T_{DAC} milliseconds. The analog output voltages are measured with an AD converter (ADC) at a T_{ADC} rate. The digital values are stored in the SUCHAI internal memory for posterior analysis.

T_{ADC} and T_{DAC} are configured by the user through command line. The payload was programmed in a way that T_{ADC} is four times smaller than T_{DAC} (Equation 1). Is important to mention that T_{ADC} and T_{DAC} could not be configured separately by the user, T_{ADC} is the only variable that is directly manipulated, and T_{DAC} is set automatically every time T_{ADC} change its value.

$$T_{DAC} = 4 \cdot T_{ADC} \tag{1}$$
$$f_c = 1 / T_{DAC} \tag{2}$$

The values of the pseudo-random number generator (RNG) are the same in every execution of the payload, the only change between executions are the milliseconds that the DAC holds each RNG value as an analog voltage, which is controlled by the T_{DAC} variable . This means that the seed of the random generator is kept constant for all executions independent of the frequency or other parameters. The input voltage resembles an uniform distribution $U_{(-1.6V,\, 1.6V)}$ between $[-1.6V,\ 1.6V]$. Figure 2 shows probability density function of the input signal measured in the laboratory.

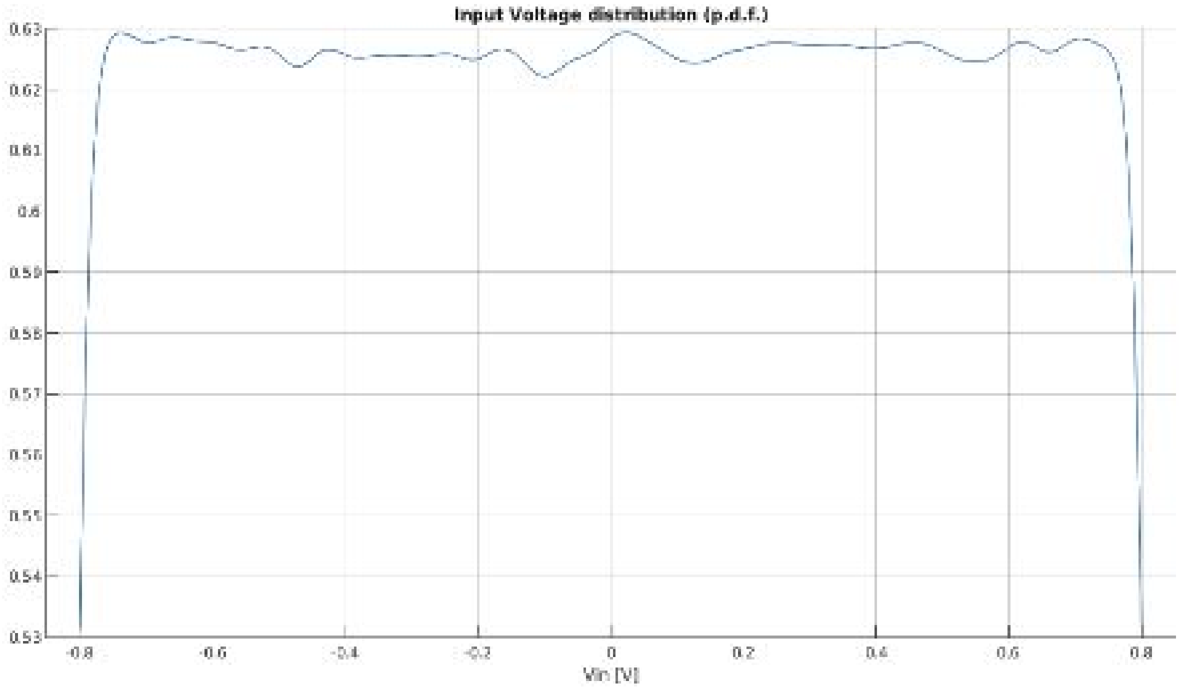


Figure 2.

The time duration of an execution is ruled by the number of values (N) that the PIC outputs to the DAC multiplied by the T_{DAC} value. The number of values generated by the DAC during the mission is fixed to $N = 16000$, therefore a total of $S = 4 * N = 64000$ samples are taken for each frequency execution (Equation 1).

Notice that the amount of values generated by the RNG and passed to the DAC is set in the firmware of the microcontroller and cannot be changed through tele-commands.

The instruments used are a 16 bit DAC -Texas Instruments DAC8551- and a 10 bit ADC included inside the PIC microcontroller. Because the RNG outputs numbers between $[0, 2^{15} - 1]$, every RNG value is amplified by two in order that the voltages in the output of the DAC are between the $[0, V_{cc}]$ range. This reduces the effective sensitivity of DAC to 15 bit.

Implementation considerations

Because of the reduced static memory of the PIC microcontroller, a buffering technique is used to store the output voltage measurements. This means that a frequency execution is implemented as a collection of smaller executions from portions of the original input signal, filling an intermediary buffer to transfer the measures to a larger memory and then resuming to the next portion of the frequency execution.

The size of the intermediary buffer is fixed to 400 samples, so every execution pauses when the buffer has 400 new samples and resumes after transferring all the samples to the larger memory (SD card). Therefore, there are a total of $\frac{64000}{400} = 160$ pauses because of this buffering to complete a whole frequency execution.

This buffering technique induces a transient response of the circuit every time that the content of the buffer is transferred to the SD card. The idea of the experiment is to analyze the steady state response of the circuit, this way one can study the effects of the hostile environment in the energy flux. Therefore any transient response impairs the experiment analysis, so a methodology to mitigate this transient response must be implemented otherwise this transient response could introduce wrong conclusions.

A simple way to mitigate the transient is to add a set of extra random values before the buffer begins to store samples. This way the circuit returns to a stationary state before the ADC begins to fill the buffer and all samples inside the buffer are taken from the stationary state. To reduce the transient even more, the DAC excites the circuit with the mean value of the input signal distribution when the a SD card transfer is about to start. This way any deviation introduced by the transient response is reduced since the input voltage is uniformly distributed.

This ideas are shown in Figure 3. where the first graph shows the input/output voltages in real time, the second graph shows only the voltages measured and the third graph shows the theoretical or expected voltages.

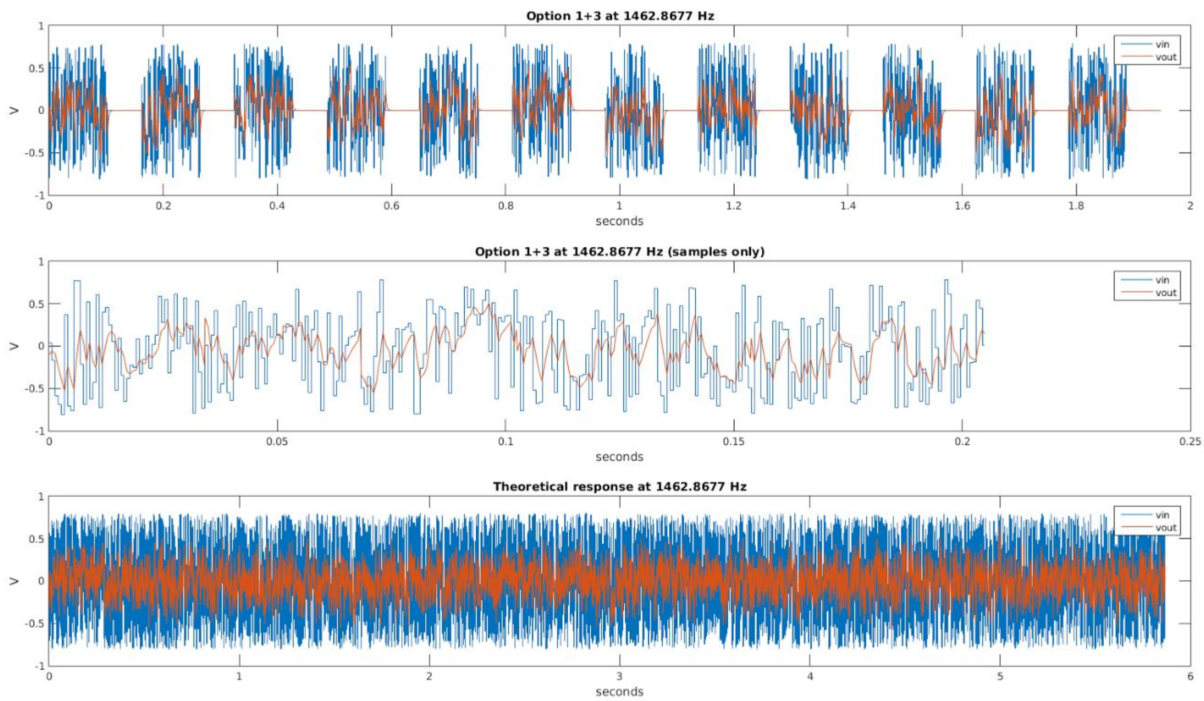


Figure 3.

Execution through Commands

SUCHAI payloads can communicate with earth station through a command interface. The structure of the commands is pretty simple because is composed of a *text header plus a set of integers*, for example:

`exe_cmd 0xABCD 12345`

This example means to execute the command **0xABCD** with the number **12345** as argument. The physics experiment has several commands available that handles its execution and are used to set the frequency f_{signal} and the seed of the RNG. A list of the commands available for the earth station is shown in Table 1.

Command	ID	Argument	Description
pay_init_expFis	0x6027	Not used.	Reset the segment of the data repository reserved for the experiment and the iteration variables used in the experiment execution.
pay_take_expFis	0x6028	Not used.	Executes the experiment with the parameters that were previously set.
pay_stop_expFis	0x6029	Not used.	Ask if a execution has been completed.
pay_adhoc_expFis	0x602A	Not used.	(EARTH only) Executes a fixed set of frequency-seed pairs.
pay_set_seed_expFis	0x602B	uint16_t seed The seed used by the RNG.	Sets the seed used by the pseudo-random number generator.
pay_set_adcPeriod_expFis	0x602C	uint16_t adcPeriod The register value that sets the ADC sampling rate.	Sets a register with the sampling period of the ADC. This period sets the frequency of the experiment by Equation 1- 3.
pay_print_seed	0x602D	uint16_t seed The seed to be printed.	Prints the integer numbers generated by the RNG with a specific seed, but only the ones that after are sampled and stored in the data repository.
pay_print_seed_full	0x602E	uint16_t seed The seed to be printed.	(DEBUG) Prints the integer numbers generated by the RNG with a specific seed (all of them without exclusion).
pay_testDAC_expFis	0x602F	uint16_t maxValue The maximum value that is sent to the DAC.	(DEBUG) Test the function of the DAC by generating a linear ramp in the input node of the circuit. The ramp goes from 0 to maxValue.
pay_testFreq_expFis	0x6030	uint16_t adcPeriod The register value that sets the ADC sampling rate.	(DEBUG) Execute the experiment with a specific frequency but with a seed value of 0. The period sets the frequency of the experiment by Equation 1- 3.

Table 1.

In a normal operation mode, only one pair of frequency/seed is executed. This is because executing two frequencies or seeds would exceed the repository capacity overwriting data from previous executions. So, during a normal operation the user send the following commands:

1. Initialize the repository segment (pay_init_expFis).
2. Sets the frequency of the input voltage (pay_set_adcPeriod_expFis).
3. Sets the seed for the pseudo-random generator (pay_set_seed_expFis).
4. Start the execution (pay_take_expFis).
5. Download the data repository (i.e. output voltage samples).
6. Print the RNG values (pay_print_seed) with the replica inside the laboratory to store the counts used by the DAC to excite the circuit (i.e. input voltage).

A linear fit between the frequency (f_c) and the adcPeriod value was made (see Figure 3). The period of the DAC was measured for differents values of adcPeriod, then the frequency was computed using Equation 2. Figure 3 shows the coefficients of the linearization.

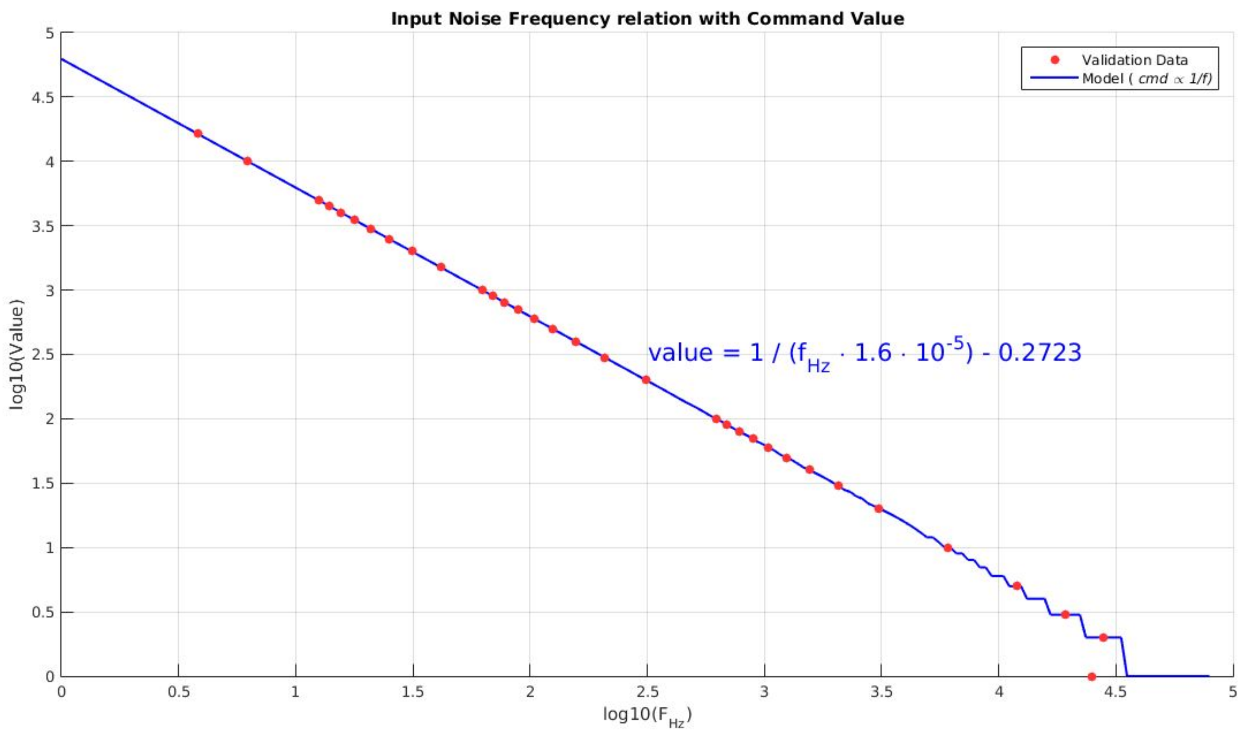


Figure 3.

$$command_{value}(f_c[Hz]) = \left[1/(f_{Hz} * 1.6 \cdot 10^{-5}) - 0.2723 \right] \quad (3)$$

For example, if an input signal of frequency $f_c = 2\,KHz$ is desired, compute the Equation 3 gives: $\left[1/(f_c * 1.6 \cdot 10^{-5}) - 0.2723 \right] = [30.97] = 31$. So, to set the experiment with a frequency $f_c = 2\,KHz$ through command line should be:

```
exe_cmd 0x602B 31
```

Is important to note that this linearization works well if the frequency f_c is less than 23KHz, after that the value, the linearization doesn't fit very well.

Eq. 3 works well if:

$$f_c < 23\ KHz \tag{4}$$

Experiment Characterization On-Earth

In order to make meaningful analysis from the data obtained during the mission, the experiment was executed and documented inside the lab. Mission data can be compared with the data obtained inside the lab and then make better analysis.

The experiment used inside the laboratory is a replica of the original experiment inside SUCHAI. Several executions and measurements were made with this replica such as varying differents parameters and trying different processing techniques. The following results comes from executions with a replica of the experiment inside the laboratory using the following setup:

1. The experiment replica has the following parameters:
    a. Voltage range $[0\ -3.3]$ Volts.
    b. $R = 1.21\ K\Omega$ and $C = 1.47\ \mu F$.
    c. Measured cutoff frequency $f_{cutoff} =\ 92\ Hz$.
    d. Pseudo-random numbers uniformly distributed as Figure 2.
    e. Input voltages values generated by the 16-bit DAC ranges between $[0, 3.3]$ V.
    f. Output voltages values acquired by the 10-bit ADC ranges between $[0, 3.3]$ V.
2. A total of **fifteen** differents frequencies executions were executed and stored.
    a. All fifteen frequencies are logarithmically spaced
        i. $f_{signal}\ =\ \{9, 15,\ 26,\ 44,\ 74,\ 125,\ 211,\ 356,\ 600\}\ Hz$ and
        ii. $f_{signal}\ =\ \{1020,\ 1725,\ 2940,\ 4710,\ 8600,\ 14650\}\ Hz$ approximately.
    b. For each frequency the experiment was executed with **three seeds**
        i. $S_{seed} = \{0\ ,\ 1000,\ 5000\}$
3. Characteristic frequency f_c and acquisition timing for the ADC is ruled by Equation 1.
4. The input and output voltages whkkere stored and used to estimate their probability density functions $v_{in}(V, f_c)$, $v_{out}(V, f_c)$ and the instant power on the capacitor $p_{inj}(mW, f_c)$ computed. The data set with the measures taken directly from the experiment will be called *Raw Data.*
5. For each one of the previous frequencies and number of points, a simulation made in Matlab/Simulink was made to contrast the empirical data with a response that doesn't have buffering issues (*Simulation data*).
6. The *Theoretical Response* was inferred by making a big simulation similar to point 5 for each frequency, but with $N \sim 3 \cdot 10^6$ input points (10^3 times bigger than experimental implementation).
7. Finally, the density estimations obtained from previous measurements were compared with different metrics:
    a. Kullback-Leibler divergence.
    b. Normal distribution parameters (mean and variance).

An analysis for each of the output variables v_{out} , p_{inj} is made to show statistical properties and the range of operation of the experiment.

Output Voltage Analysis

Figures 6, 8, 10, 12 shows the time series for the output voltage $v_{out}(t)$ and Figure 7, 9, 11, 13 shows the density function estimation for different frequencies. Notice that in all the figures there are four data sets compared:

1. ***Raw Data:*** corresponds to data described in point 4.
2. ***Simulation:*** corresponds to simulation without buffering that has the same number of samples as *Raw Data.* It's described in point 5.
3. ***Theoretical:*** corresponds to a "bigger" simulation without buffering that is described in point 6.

Notice that time series plots are only for raw data so the peaks come from the transient response discussed in section "*Implementation considerations*". Density estimations were made with a gaussian kernel implemented in Matlab using the `ksdensity()` function.

Because this RC circuit is a low pass filter, one can see that while the frequency decreases the time series for the output signal looks more like the input signal. Furthermore, for lower values of frequency the effect of buffering decreases (Figure 4,5).

Since the input signal distribution is uniform $v_{in} \sim U_{(-0.8,\, 0.8)}$ and the circuit behaves as a first order low pass filter, the expected distribution for the output voltage should be a normal distribution dependent on the frequency $v_{out}(V, f_c) \sim N(0, \sigma^2(f_c))$. The *theoretical* function estimation in Figures 7, 9, 11 and 13 proves this statement.

Lastly, the *simulation* function should be the same distribution as the *theoretical* function. Their only real difference is the number of samples they have, *simulation* has the same number of samples than the on-board experiment ($S = 64.000$) and *theoretical* has $\sim 10^3$ times more samples.

Injected Power Statistics

Similarly as for the output voltage statistics, the Figure 14, 16, 18 and 20 show the time series for the injected power and the Figure 15, 17, 19 and 21 show their density function estimates.
The *theoretical* density function appears to be symmetric for high frequencies (Figure 15 and 17) and asymmetric for lower frequencies with higher probability for positive values. The *filtered* function appears to show the same kind of symmetry.

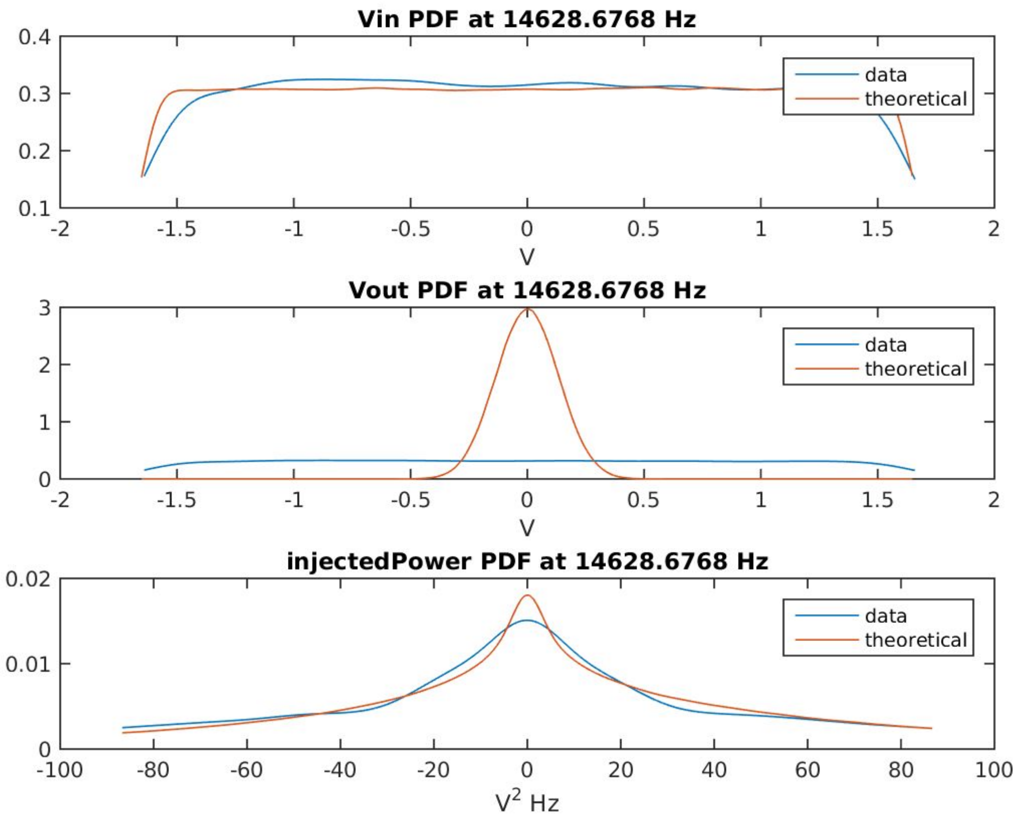
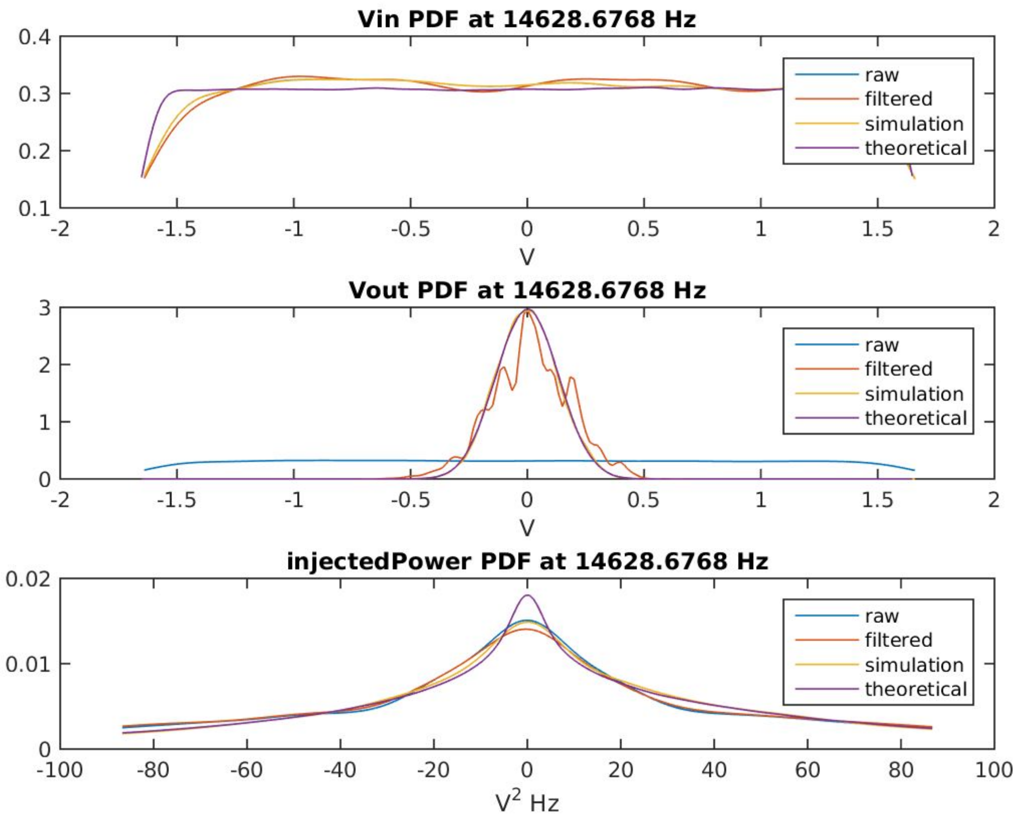

Vin PDF at 14628.6768 Hz
data
theoretical
V
Vout PDF at 14628.6768 Hz
data
theoretical
V
injectedPower PDF at 14628.6768 Hz
data
theoretical
V^2 Hz

Figure 14



Vin PDF at 14628.6768 Hz
V
raw
filtered
simulation
theoretical
Vout PDF at 14628.6768 Hz
V
raw
filtered
simulation
theoretical
injectedPower PDF at 14628.6768 Hz
V^2 Hz
raw
filtered
simulation
theoretical

Figure 15

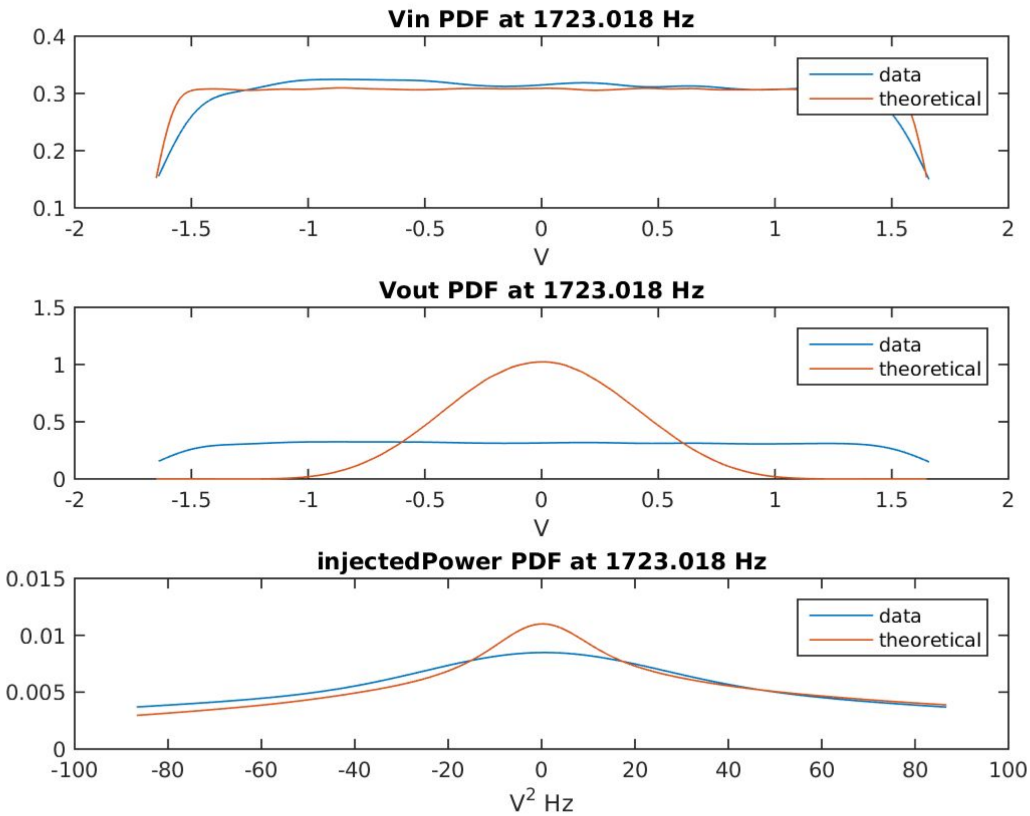
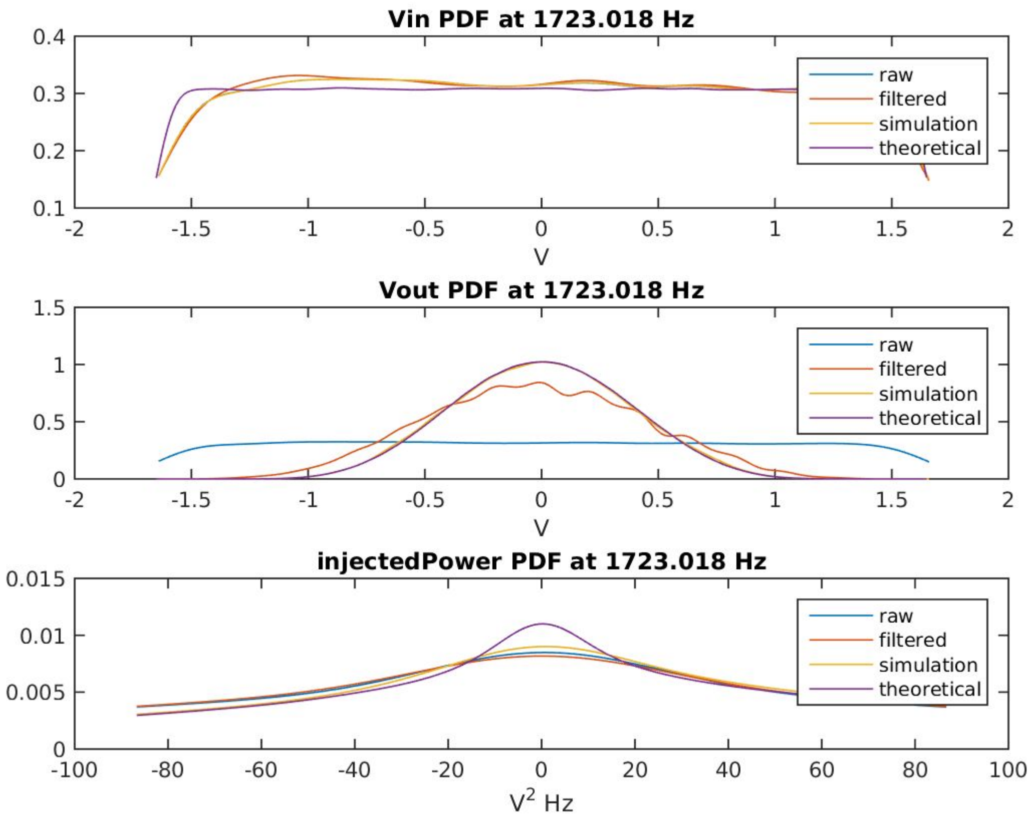
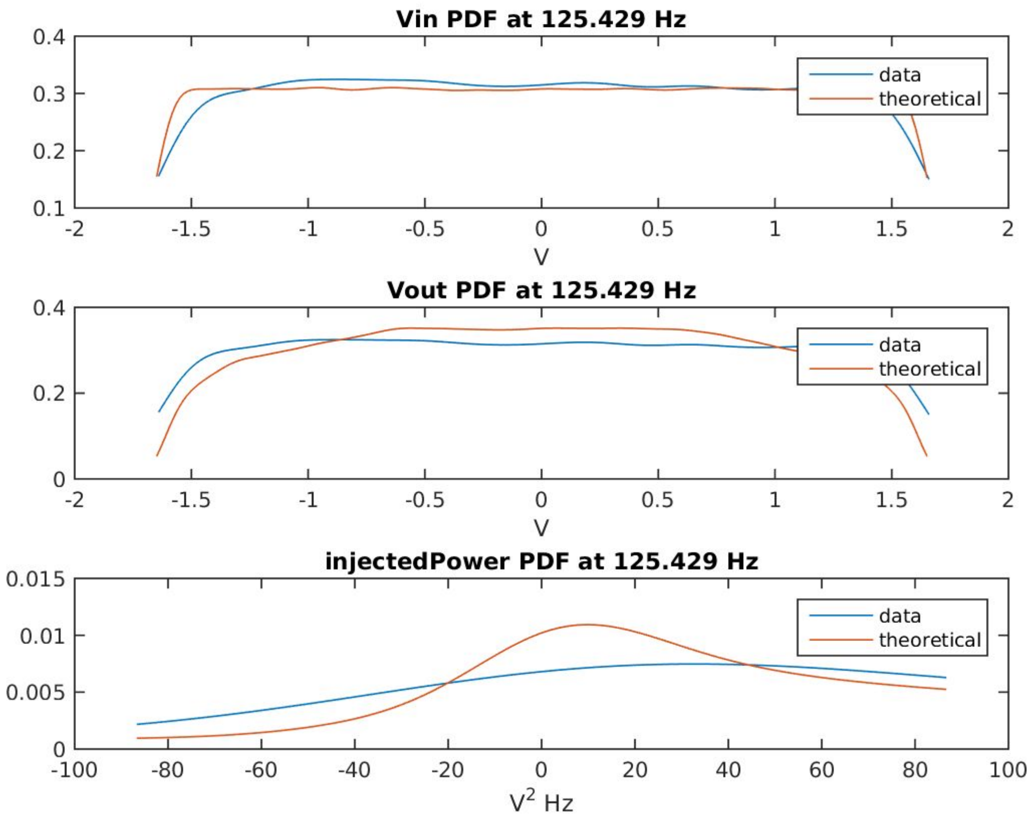

Vin PDF at 1723.018 Hz
data
theoretical
V
Vout PDF at 1723.018 Hz
data
theoretical
V
injectedPower PDF at 1723.018 Hz
data
theoretical
V^2 Hz

Figure 16



Vin PDF at 1723.018 Hz
raw
filtered
simulation
theoretical
V
Vout PDF at 1723.018 Hz
raw
filtered
simulation
theoretical
V
injectedPower PDF at 1723.018 Hz
raw
filtered
simulation
theoretical
V2 Hz

Figure 17



Vin PDF at 125.429 Hz
0.4
0.3
0.2
0.1
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
V
data
theoretical
Vout PDF at 125.429 Hz
0.4
0.2
0
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
V
data
theoretical
injectedPower PDF at 125.429 Hz
0.015
0.01
0.005
0
-100
-80
-60
-40
-20
0
20
40
60
80
100
V2 Hz
data
theoretical

Figure 18

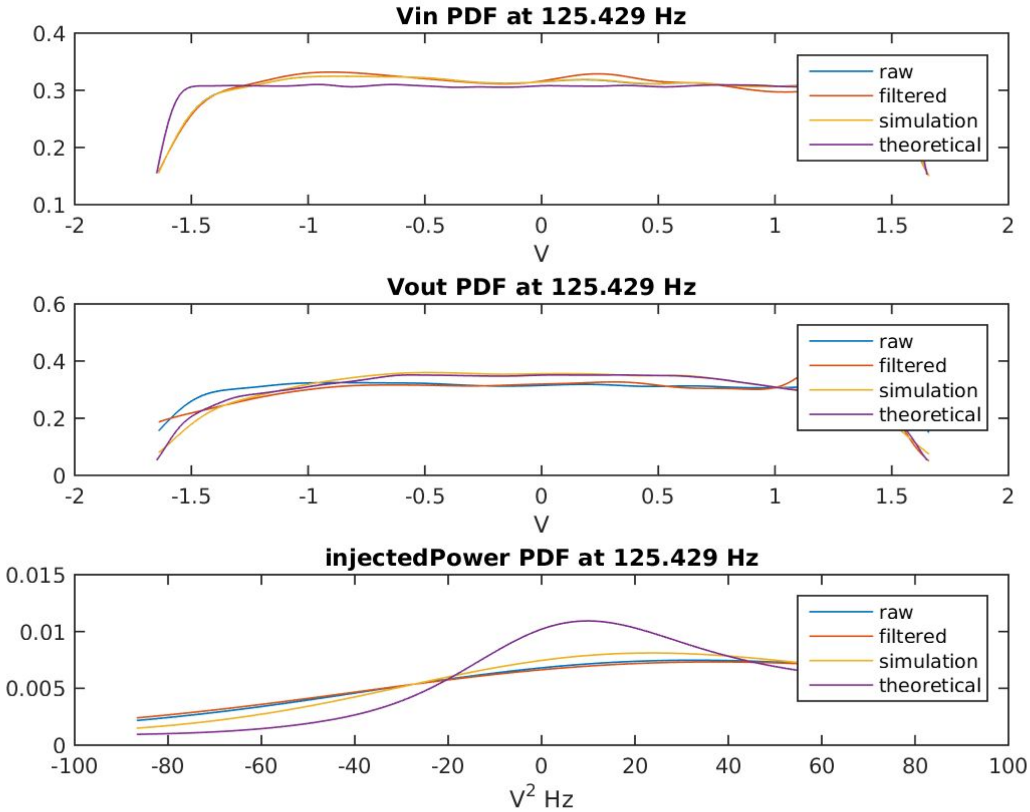

Vin PDF at 125.429 Hz
raw
filtered
simulation
theoretical
V
Vout PDF at 125.429 Hz
raw
filtered
simulation
theoretical
V
injectedPower PDF at 125.429 Hz
raw
filtered
simulation
theoretical
V2 Hz

Figure 19

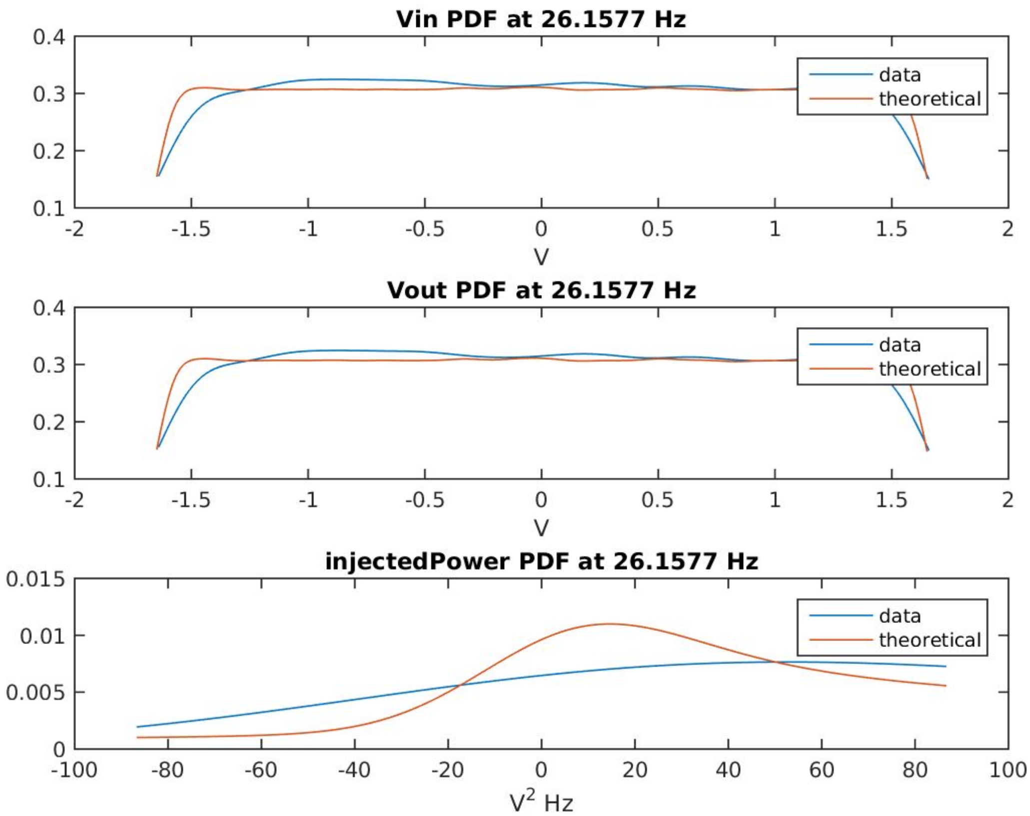

Vin PDF at 26.1577 Hz
data
theoretical
V
Vout PDF at 26.1577 Hz
data
theoretical
V
injectedPower PDF at 26.1577 Hz
data
theoretical
V^2 Hz

Figure 20

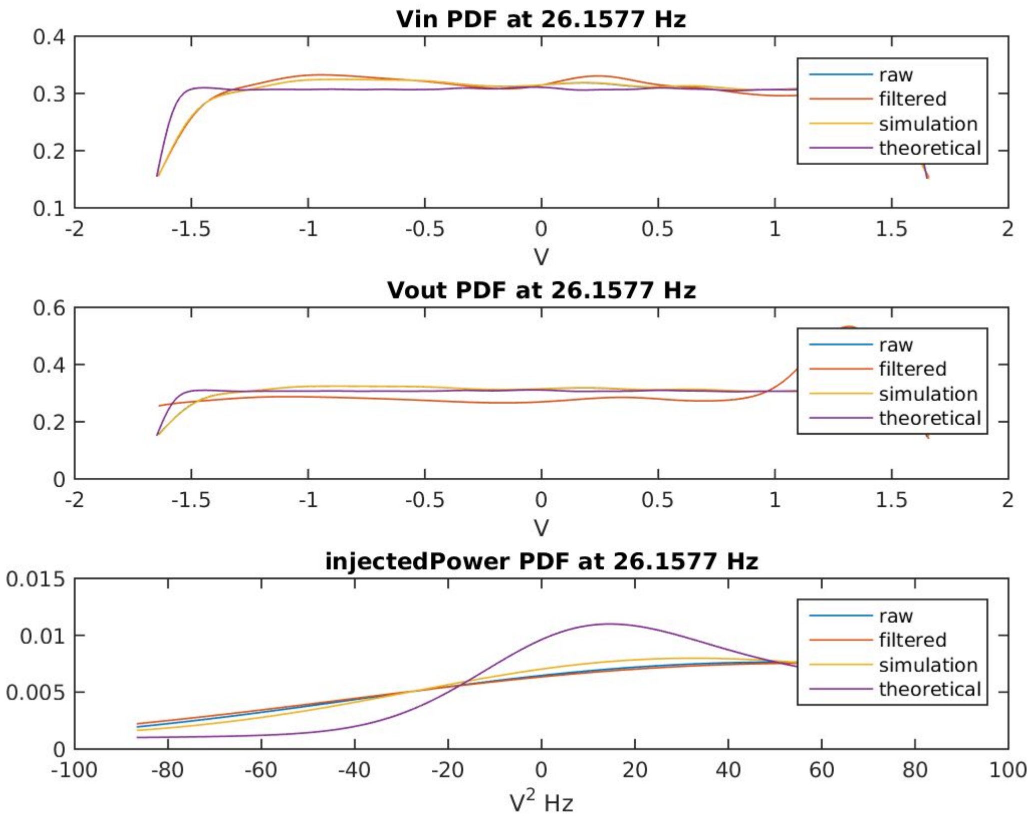


Figure 21

Inferred Density Functions Analysis

Depending on the execution frequency f_c, both variables v_{out}, p_{inj} have density functions pretty similar to the *theoretical* and *simulation* functions. A method to quantify this similarities is needed.

Assuming that the empirical distribution has a known distribution the simplest way to compare with the *theoretical* function is computing the parameters of the empirical distribution and compare with the parameter of the *theoretical* distribution. Given that the input signal distribution is always the same, the output voltage seems to distribute normal for a range of frequencies $v_{out}(V, f_c) \sim N(0, \sigma^2(f_c))$ and a gaussian fit could be used to measure the variance as dependant on the frequency $\sigma^2(f_c)$.

In the other hand $p_{inj}(mW, f_c)$ doesn't distribute normal or in a known function, so other metrics must be used.

The *Kullback-Leibler Divergence* (KL Div) is introduced as a metric to quantify the difference between two arbitrary distributions. This is a metric from information theory that helps to analyze the differences between the empirical results obtained through the replica and the simulated/theoretical model.

So, the metrics used for each distribution are the following:

1. For $v_{out}(V, f_c)$
    a. Kullback-Leibler divergence.

b. Normal distribution parameters (mean and variance).

2. For $p_{inj}(mW, f_c)$

   a. Kullback-Leibler divergence.

When computing the KL Divergence for each distribution $P_{v_{out}}(V)$ and $P_{p_{inj}}(mW)$, the comparisons made are:

1. ***Raw Data* vs *Theoretical Response***: This shows how the experimental results differs from the expected or theoretical response..
2. ***Raw Data* vs *Simulation (Simulink)*:** This shows how the experimental results differs from an ideal implementation with the same number of samples.

Figure 22-25 shows a frequency dependent graph that shows this comparisons using the KL Divergence for the output voltage distribution.

Comparing Figure 22 and 23, one can see that the transient response impairs drastically the density function for high frequencies $f_c > 1\ KHz$, that is why the filtering mitigates this effect having a nearly constant difference between the empirical data versus the theoretical distribution.

Figure 24 shows how for high frequencies the real implementation differs from an ideal implementation of the experiment. This helps to decide the upper limit for frequencies executed during the mission.

Lastly, Figure 25 shows the effect of filtering in the function estimation. Clearly, for low frequencies the functions are pretty similar to unfiltered case, but for high frequencies the filter helps much better to the density estimation.

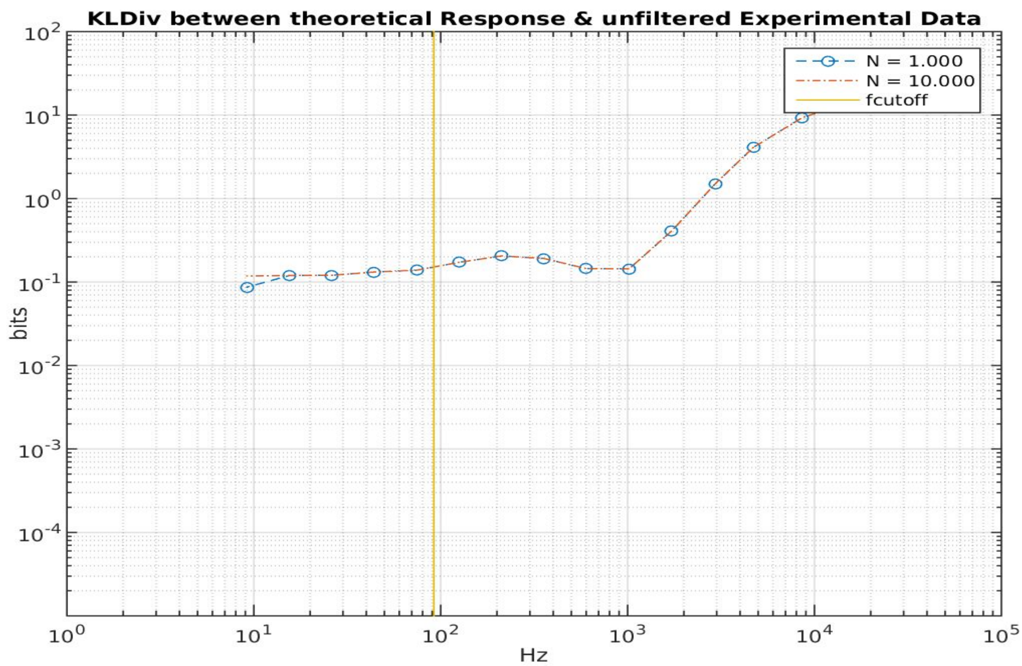


Figure 22

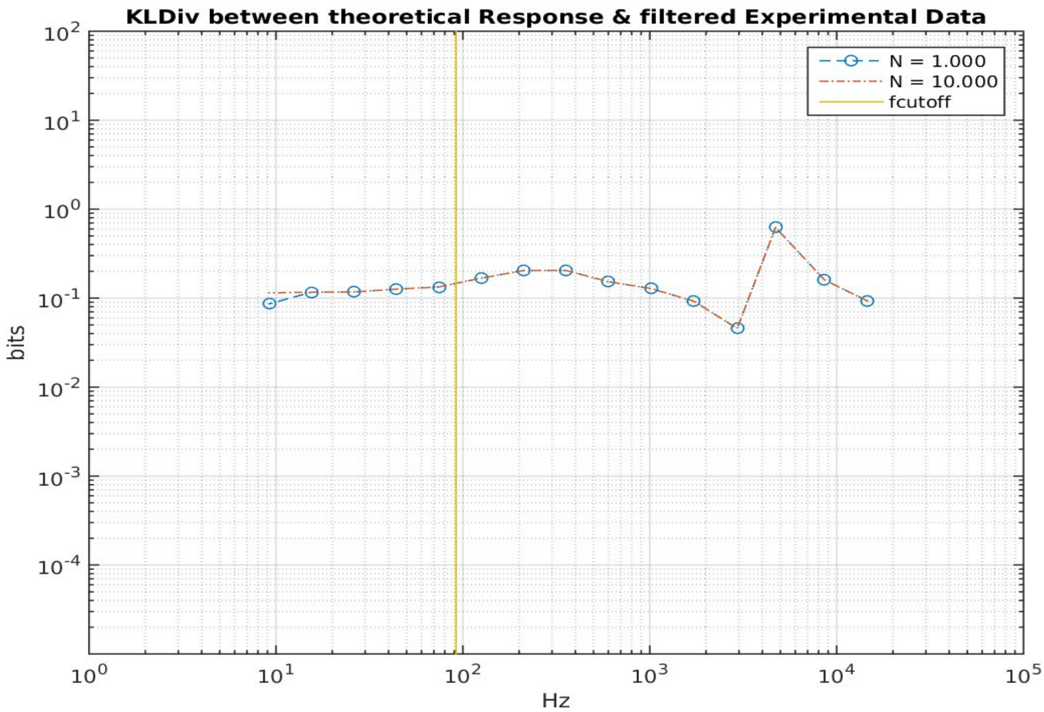


Figure 23

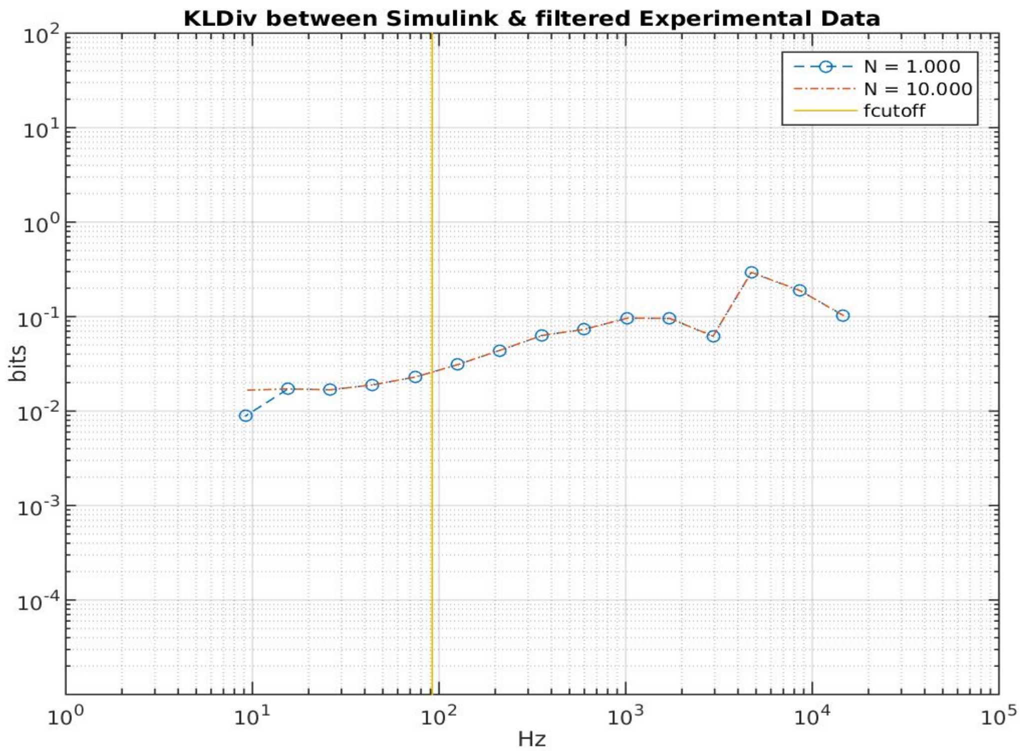


Figure 24

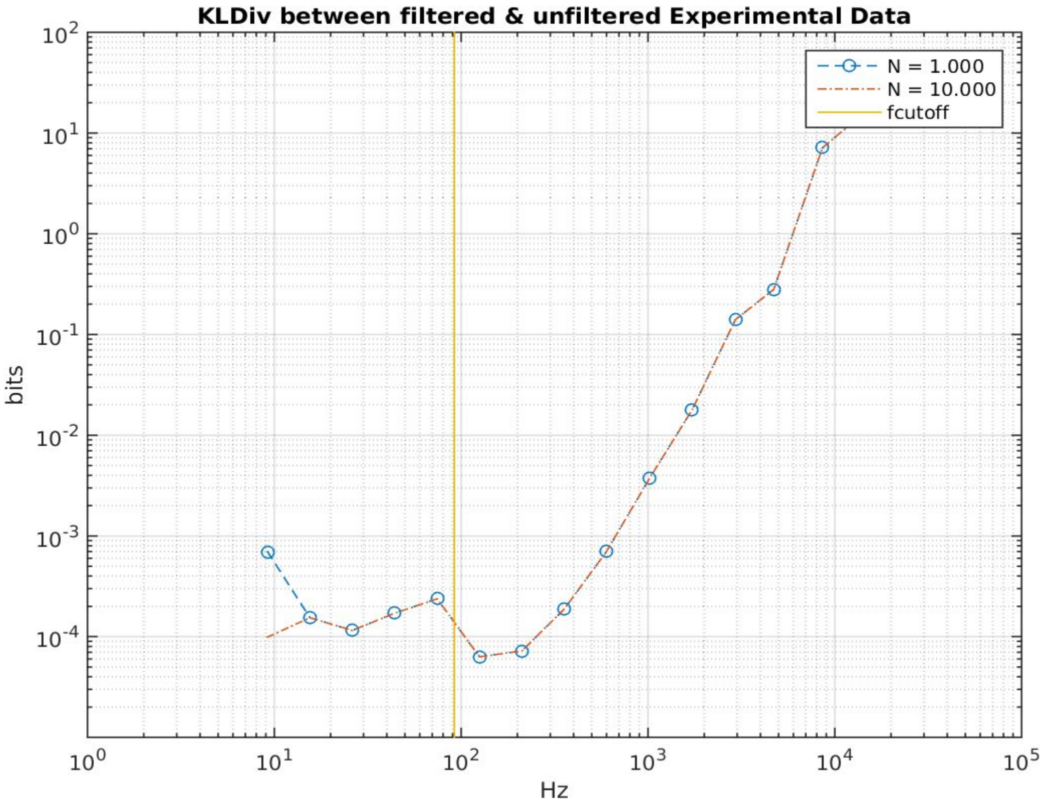


Figure 25

Figure 26 shows the variance dependency over the frequency was inferred from the output voltage distributions estimated, and the equation fitting:

$$\sigma^{2}(f_c) = \frac{a*f_c}{(f_c+b)^{2}} \tag{6}$$

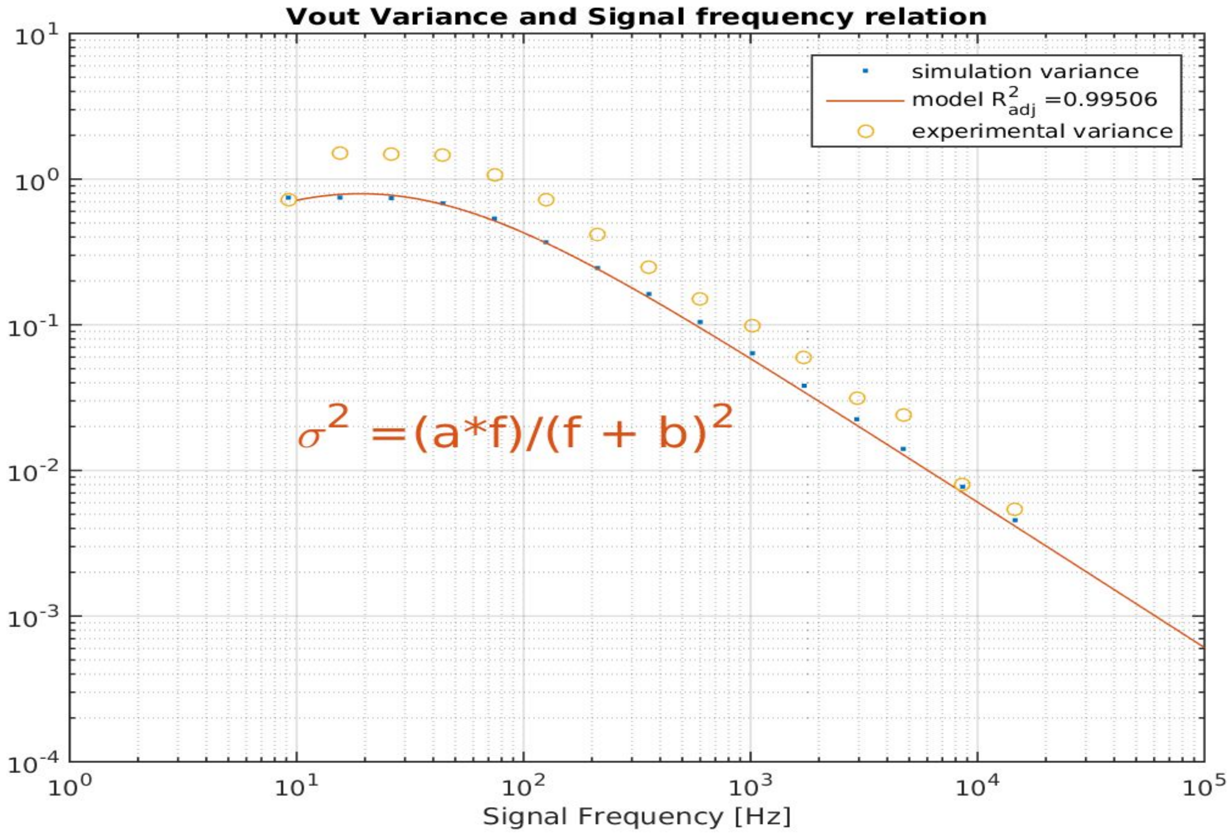


Figure 26

Now the variance for the filtered data versus the expected variance is shown (Equation 6, Figure 27). Unlike others metrics, the fitted variance improves with higher frequencies, this is because for lower frequencies v_{out} doesn't adjust well to a normal distribution (see Figure 11 and 13) compared to high frequencies (see Figure 7 and 9). Because of this, the variance appears to be less convenient than KL divergence since the v_{out} does not distribute normal for every frequency.

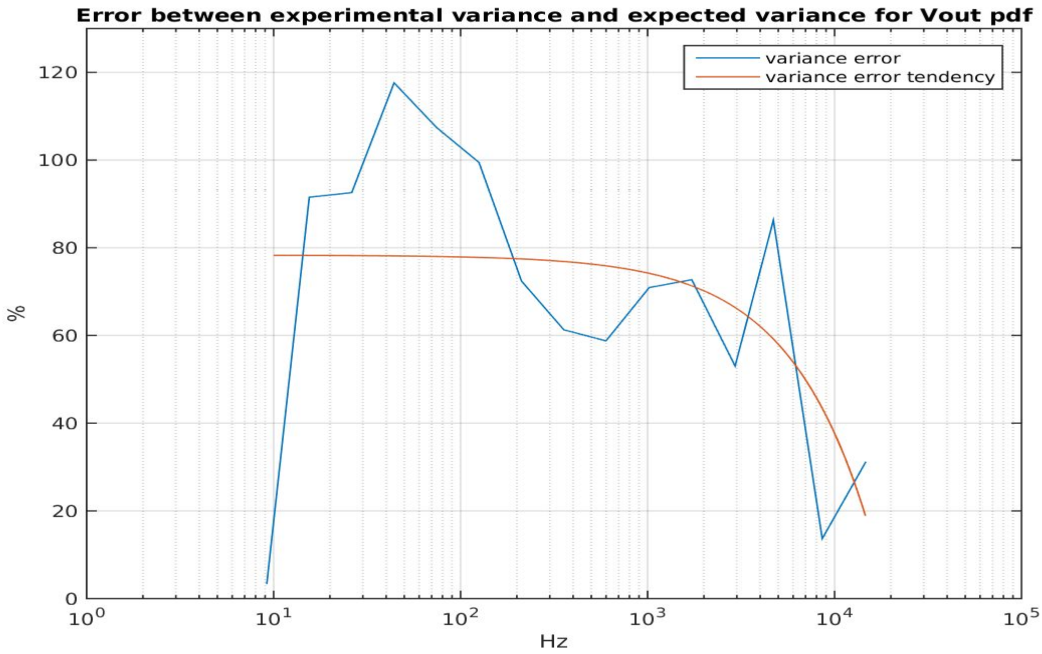


Figure 27

Figure 28 - 32 shows the same plots, but for the injected power distribution:

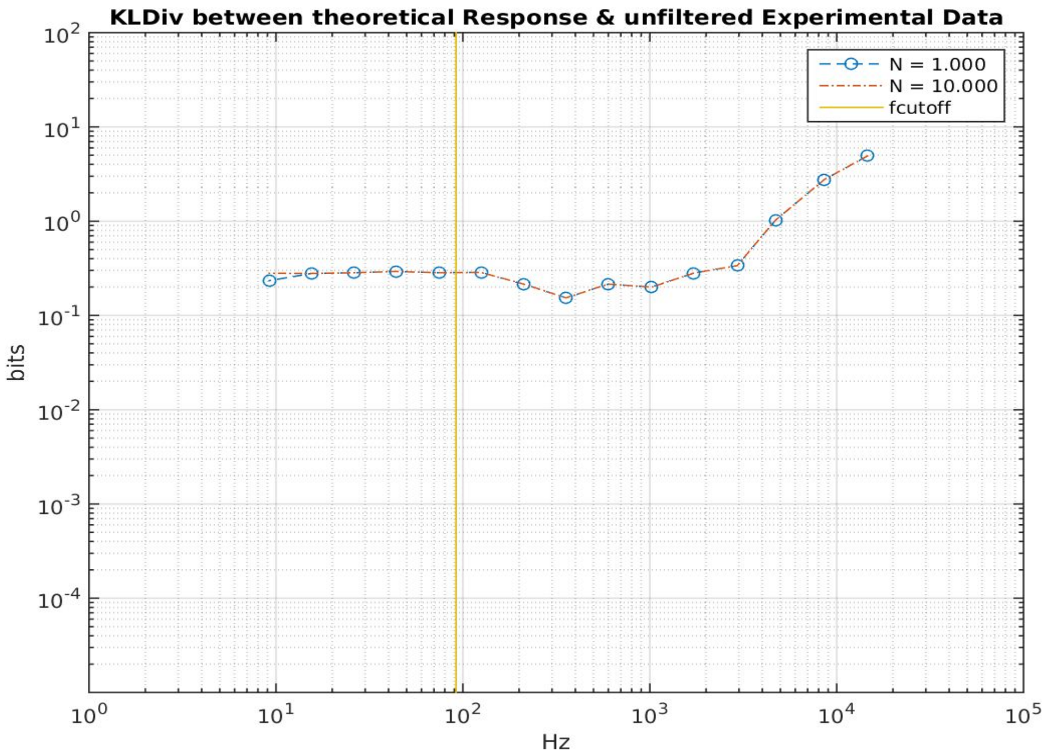


Figure 28

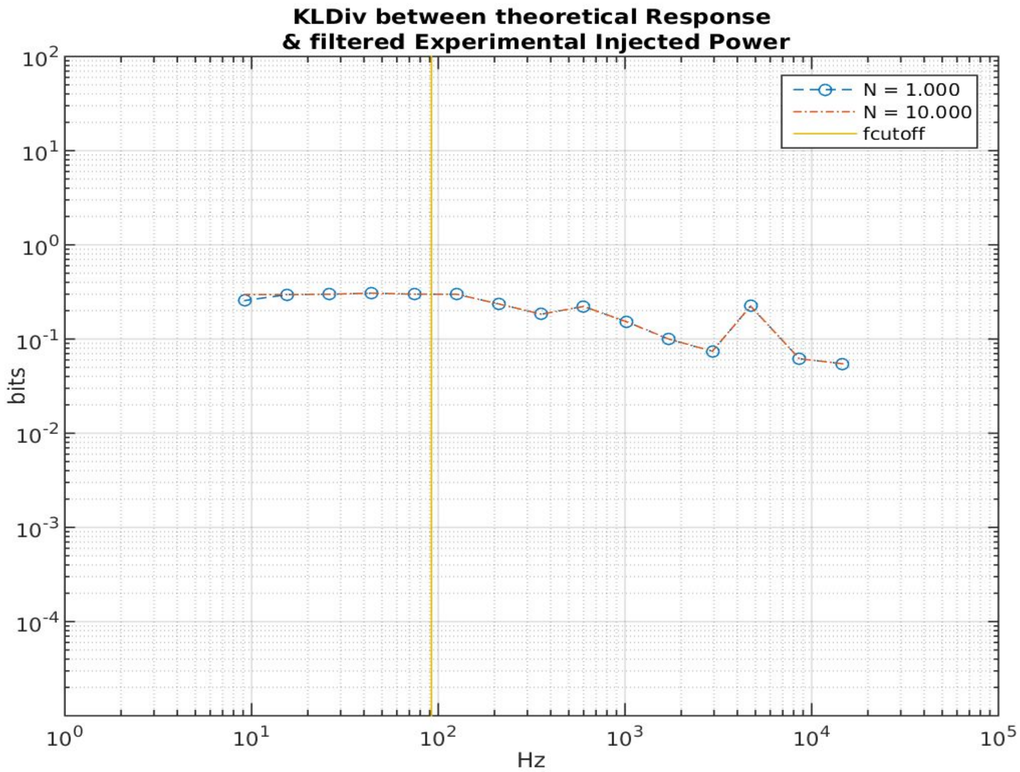


Figure 29

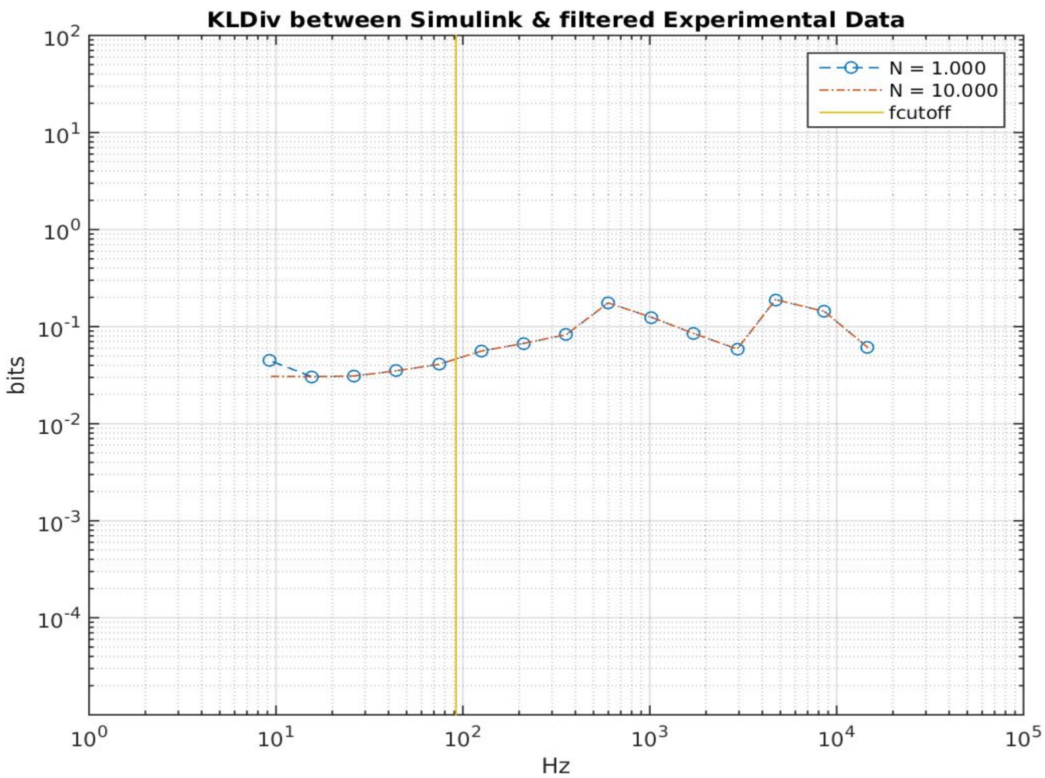


Figure 30

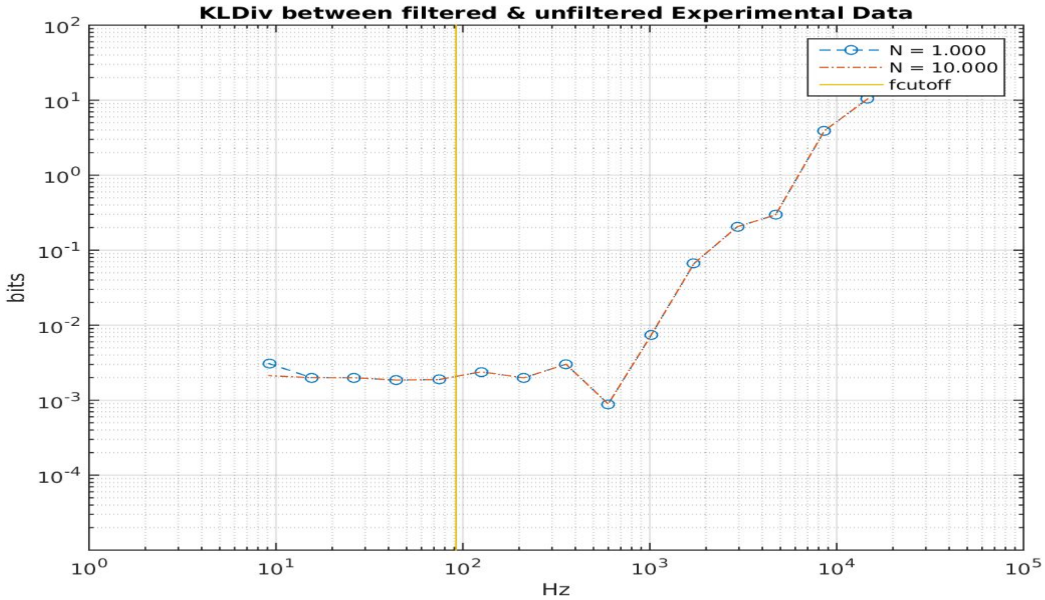


Figure 31

Finally, notice that a greater number of samples $N/S = 10.000/40.000$ doesn't seem to add more information than the on-board experiment ($N/S = 1.000/4.000$), since all graphs of KL Divergence are the same for each case.

Finally, a maximum range of frequency operation must be set. Looking the *filtered-theoretical* comparison of KL Div for v_{out} and p_{inj}, a maximum frequency of $max(f_c) = 2.5\,KHz$ should work. So, the experiment range of operation should something like:

$$f_c \in [10, 2500]\,Hz \tag{7}$$

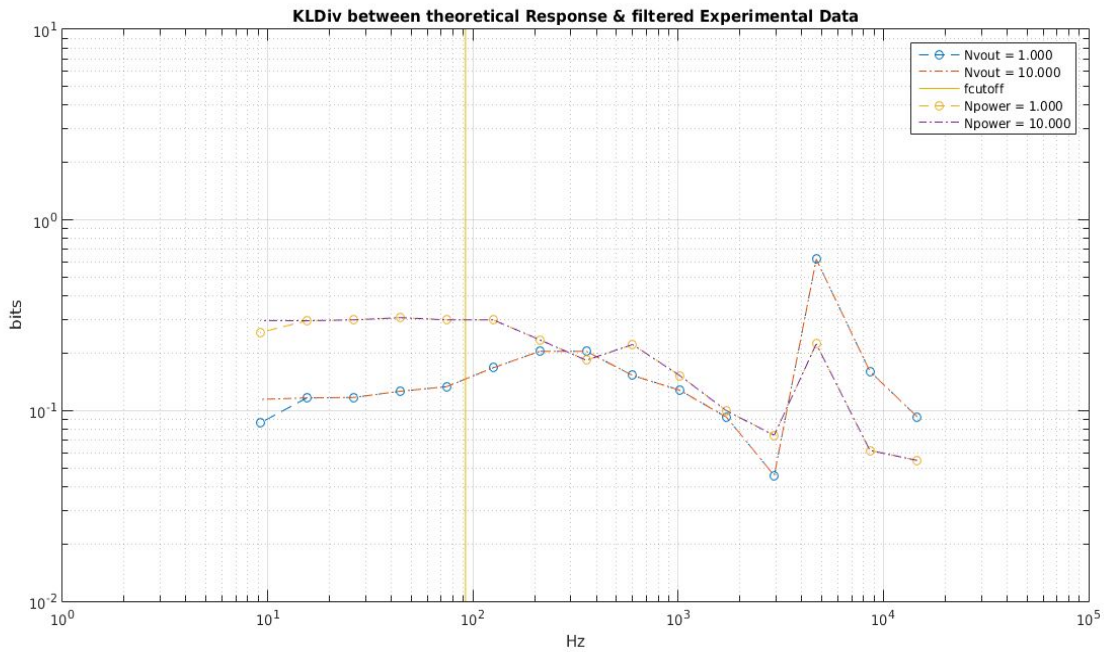


Figure 32

Conclusions

A full documentation of the physics experiment aboard the SUCHAI 2/3 cubesat has been presented. Besides the same experiment was mounted in SUCHAI 1 this document shows the same payload, but with better transient response mitigation than the previous one.

The main objective of the experiment was discussed in the beginning of this document, as well as its expected results were documented. Normal execution of the experiment through earth-station commands was explained step by step.

Analysis of the experiment was made inside the laboratory with a replica, executing several times and changing fixed parameters such as: range of frequencies and different seeds. A simple analysis centered in information theory metrics such as the Kullback Leibler Divergence has been made to find the maximum range of frequency operation for this experiment and the effect of the amount of samples in the density function estimation.

The density function estimation inside the cubesat is the best option to decongest the communication with the earth station, but because of the reduced computation power on the satellite this choice was discarded. Also, modularization of the code and more intuitive commands are considered as future work for this payload.

References

[1] Falcón, C., & Falcon, E. (2009). Fluctuations of energy flux in a simple dissipative out-of-equilibrium system. Physical Review E, 79(4), 041110.