



Prepared by

Institute for Research in Technology (IIT)
ICAI Engineering School
Comillas Pontifical University

for

CRISA

in the context of the project

**“ACTIVE VIBRATION CANCELATION FOR A
CRYO-COOLER COMPRESSOR IN SPACE
APPLICATIONS”**

[Details of the vibration-cancellation algorithm]

October 2011

Version: 1.0

Ref.IIT: IIT-11-179IC

Authors: Aurelio García Cerrada and Juan Luis Zamora Macho

Ownership and Responsibility

The copyright of this work corresponds to the members of the research team, which must be referenced in any use of its results.

The conclusions and opinions expressed in this report belong exclusively to the authors, and do not compromise to any extent Comillas Pontifical University or any of its Centres, Institutes, Professors or Researchers.

CONTENTS

1. PROBLEM DESCRIPTION.....	3
2. ABOUT THIS DOCUMENT	5
3. ACTIVE VIBRATION-CANCELLATION ALGORITHM. FUNDAMENTALS	5
3.1 INTRODUCTION	5
3.2 VIBRATION CANCELATION USING OPTIMIZATION WITH GAUSS-NEWTON ALGORITHM.....	5
3.3 COMMAND-SIGNAL GENERATION	10
3.4 IMPLEMENTATION DETAILS.....	11
3.4.1 DFT implementation.....	11
3.4.2 Computation of the real and imaginary parts of the position reference.....	14
3.4.3 Perturbing real part of the command-signal components	16
3.5 PARAMETERS OF THE VIBRATION-CANCELLATION ALGORITHM	23
3.6 ALGORITHM FLOW FOR VIBRATION CANCELLATION.....	25
4. DESCRIPTION OF THE TEST-RIG BUILT BY CRISA.....	27
5. FURTHER DETAILS OF THE IMPLEMENTATION OF THE VIBRATION-CANCELLATION ALGORITHM IN THE TEST-RIG	29
6. EXPERIMENTAL RESULTS OF THE VIBRATION-CANCELLATION ALGORITHM... 	30
6.1 VIBRATION-CANCELLATION WITHOUT TEMPERATURE CONTROL	30
6.2 TESTING THE VIBRATION-CANCELLATION ALGORITHM WITH TEMPERATURE CONTROL	33

1. Problem description

The type of cryo-coolers to which this study is oriented consists of two symmetrically-placed linear motors as shown in Fig.1.

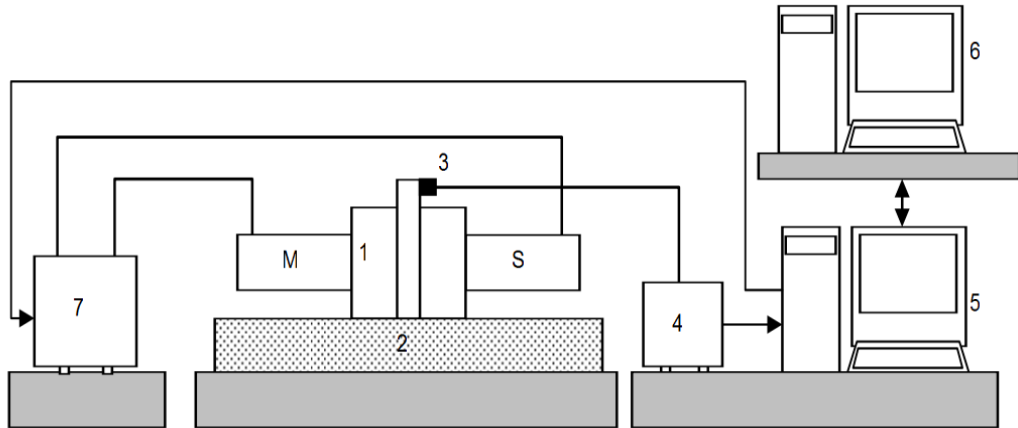


Fig. 1: Test rig for a cryo-cooler compressor prototype built by CRISA¹

The position of one of the linear motors (Master or main motor, M in Fig. 1) is closed-loop controlled to follow a reference signal which is calculated by the cold-tip temperature controller. This reference signal is a known-and-constant frequency and variable-amplitude sinusoidal which is shaped by the cold-tip reference temperature and the actual working conditions. The position of the other linear motor (Slave or auxiliary, S in Fig. 1) is also closed-loop controlled. Its reference is synchronized with the master's reference so that its movement opposes the master's in order to cancel the structure vibration caused by the master. In the solution finally implemented, the position controllers in both motors contain an inner closed-loop current control which calculates the required motor voltages. These voltages are applied using power electronic converters (7 in Fig. 1). Ideally, the movement of the symmetrically placed motors would result in no vibration in the compressor. However, any small asymmetry can cause vibration which could disturb the operation of close devices. The fundamental vibration frequency coincides with the movement frequency of the motor and harmonics of this frequency will also be present in the system vibration. When active vibration control is required, the position command signal for the slave motor is complemented by a vibration-cancellation algorithm. Vibration can be measured by one or more accelerometers. This approach is depicted in Fig. 2.

¹ This figure and Figs. 7, 8, 9, have been adapted from the document CRISA S3-TN-CRS-SL-00140.

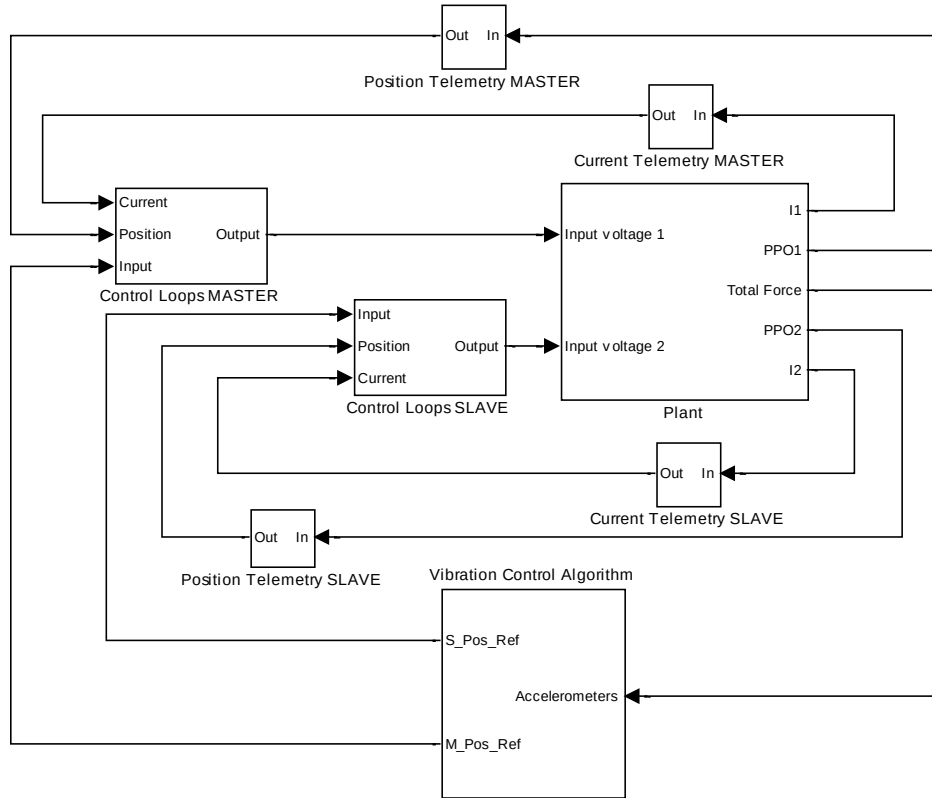


Fig. 2: Active vibration control in a cryo-cooler compressor. Overview

In Fig. 2:

- *Input voltage1* and *Input voltage2* are motor1 (MASTER) and motor2 (SLAVE) input voltages, respectively.
- *I1* and *I2* are motor1 and motor2 currents, respectively. They are fed back using current transducers.
- *PPO1* and *PPO2* are motor1 and motor2 position measurements which are fed back using position transducers.
- *Total Force* is the net force applied on the structure which produces its vibration. The acceleration produced by this force is measured with accelerometers.
- *M_Pos_Ref* and *S_Pos_Ref* are the position-reference signals for motor1 and motor2, respectively. The former is calculated by the temperature controller (which is not presented in Fig. 2), the latter is obtained complementing the former with the output of the vibration controller.
- *Accelerometers* are the acceleration measurements to be fed back. Accelerometers give time-domain signals.

2. About this document

This document describes an active vibration-cancellation algorithm for a cryo-cooler compressor, in detail. This document contains:

- A detailed explanation of the vibration-cancellation algorithm proposed to be implemented (Section 3), including implementation details.
- A brief description of the test-rig built by CRISA for testing (Section 4).
- A description of further implementation details which apply to the test-rig built by CRISA (Section 5).
- Experimental results obtained in the test-rig (Section 6).

3. Active vibration-cancellation algorithm. Fundamentals

3.1 Introduction

This section describes the theoretical fundamentals behind the active vibration-cancellation algorithm proposed to be used in a cryo-cooler. In this algorithm, it is not necessary to have a detailed dynamical model of either the vibration production mechanism or the influence of the slave motor movement (controlled input). The way in which the slave motor movement affects vibration is estimated on-line. This approach relays on the assumption that the vibration is a stationary phenomenon that will not change if the pattern of the slave motor movement is not changed.

3.2 Vibration cancelation using optimization with Gauss-Newton algorithm

To start with, it has been assumed that vibration is caused by mechanical asymmetries in the compressor and that vibration is measured using several accelerometers placed in various places of the device structure. The algorithm uses the measurements to calculate (without a pre-known model) how the position-reference signal for the slave motor has to be modified to achieve the established goal which takes the form of a cost function to be minimized, recursively, using Gauss-Newton algorithm.

Goal: To minimize, for each harmonic, the sum of the squares of the amplitudes measured by each accelerometer.

The harmonics (from n_1 to n_n) to which the minimization process is applied have to be previously defined by the user (harmonics from 1st to 7th, for example). Therefore, n cost functions are minimized, each one (V_{n_i}) corresponding to a different harmonic (n_i).

$$V_{n_i} = \sum_{n_j=1}^{n_s} (A_{yn_i n_j}^2 + B_{yn_i n_j}^2) \quad (1)$$

where $A_{yn_i n_j} + jB_{yn_i n_j}$ is the complex amplitude of harmonic n_i measured by accelerometer n_j and n_s is the total number of accelerometers to be used. Harmonics from n_1 to n_n for each accelerometer n_j are calculated as complex numbers such as $(A_{yn_i n_j} + jB_{yn_i n_j})$, applying a DFT² to the time-domain data. This processes can be written as:

$$\begin{bmatrix} A_{yn_1 1} + jB_{yn_1 1} & \cdots & A_{yn_1 n_s} + jB_{yn_1 n_s} \\ \vdots & \ddots & \vdots \\ A_{yn_n 1} + jB_{yn_n 1} & \cdots & A_{yn_n n_s} + jB_{yn_n n_s} \end{bmatrix} = \frac{2}{N} \begin{bmatrix} 1 & e^{-\frac{2\pi j}{N} N_C n_1} & \cdots & e^{-\frac{2\pi j}{N} (N-2) N_C n_1} & e^{-\frac{2\pi j}{N} (N-1) N_C n_1} \\ \vdots & \cdots & \cdots & \cdots & \vdots \\ 1 & e^{-\frac{2\pi j}{N} N_C n_n} & \cdots & e^{-\frac{2\pi j}{N} (N-2) N_C n_n} & e^{-\frac{2\pi j}{N} (N-1) N_C n_n} \end{bmatrix} Y_k \quad (2)$$

Where N is the total number of data used in the DFT, N_C is the number of cycles to be used for the DFT and Y_k is the N -data buffer to be used for the DFT. It is worth pointing out that all the exponentials in (**Error! No se encuentra el origen de la referencia.**) are constant values and can be calculated before the implementation stage. Data sampling period is t_s seconds and must be chosen so that a cycle of the fundamental vibration frequency contains and integer number of samples.

The way in which vibration changes due to changes in the position-reference signal (command) has to be quantified using sensitivities. For each harmonic to be cancelled (n_i) and for each accelerometer (n_j), one has to calculate 4 sensitivities:

$$\frac{\partial A_{yn_i n_j}}{\partial A_{un_i}}, \frac{\partial A_{yn_i n_j}}{\partial B_{un_i}}, \frac{\partial B_{yn_i n_j}}{\partial A_{un_i}}, \frac{\partial B_{yn_i n_j}}{\partial B_{un_i}}$$

using the finite-difference method³. Subscript u is used for input-related (slave command) variables while subscript y is used for output-related (vibration) variables. The finite-difference method is applied as follows:

- The real part of the harmonics of the command signal (A_{un_i}) are incremented.
- The changes on the real ($A_{yn_i n_j}$) and imaginary ($B_{yn_i n_j}$) parts of the vibrations recorded are evaluated for each accelerometer.

² Discrete Fourier Transform

³ Sensitivities give information of the “vibration-production mechanism” without the need of using an analytical model.

- The imaginary part of the harmonics (B_{un_i}) of the command signal are incremented.
- The changes on the real ($A_{yn_i n_j}$) and imaginary ($B_{yn_i n_j}$) parts of the vibrations recorded are evaluated for each accelerometer.
- The sensitivities are calculated using finite-differences with the data obtained in the previous steps.

$$\begin{aligned}
\frac{\partial A_{yn_i n_j}}{\partial A_{un_i}} &\cong \frac{\Delta A_{yn_i n_j}}{\Delta A_{un_i}} \\
\frac{\partial A_{yn_i n_j}}{\partial B_{un_i}} &\cong \frac{\Delta A_{yn_i n_j}}{\Delta B_{un_i}} \\
\frac{\partial B_{yn_i n_j}}{\partial A_{un_i}} &\cong \frac{\Delta B_{yn_i n_j}}{\Delta A_{un_i}} \\
\frac{\partial B_{yn_i n_j}}{\partial B_{un_i}} &\cong \frac{\Delta B_{yn_i n_j}}{\Delta B_{un_i}}
\end{aligned}$$

(3)

When calculating sensitivities, it is assumed that the harmonic components of the vibration are decoupled. This makes it possible to carry out all the calculations simultaneously with an important time saving. Although this assumption might not be true, no major complication has been observed in the simulation and experimental tests in this project. In fact, only two conflictive situations can be foreseen:

- If changes in the amplitude of one harmonic in the command signal can produce vibration at lower frequencies (sub-harmonics).
- If changes in the amplitude of one harmonic in the command signal cannot produce any vibration change at that frequency.

If a change in a harmonic of the command signal only affects vibration at the same frequency or at higher frequencies, the optimization process will converge successfully.

Each time the command-signal contents are changed, the algorithm has to wait before measurements are taken in order to make sure that steady state is reached. This time is implemented as an integer number of cycles of the fundamental frequency (N_{lat}) and is called “latency”. Therefore the DFT results are ready every T_s , seconds⁴:

$$T_s = (N + N_{lat} \cdot N_p) t_s \quad (4)$$

⁴ The algorithm samples data with a “fast” sampling frequency ($f_s=1/t_s$) and the DFT results are available with a “slow” sampling frequency ($1/T_s$).

with, N_p = number of points taken per cycle of the fundamental frequency.

The user will have to set the values for the following parameters:

- The size of the real- and imaginary-part changes for sensitivity calculation (ΔA_u and ΔB_u , respectively),
- The number of cycles to be used for the DFT calculation and
- The number of latency cycles.

The cost function in (1) to be minimized by the Gauss-Newton algorithm can also be written as a scalar product of two vectors (one row vector and one column vector):

$$\begin{aligned}
 V_{n_i}(A_{un_i}, B_{un_i}) &= \sum_{n_j=1}^{n_s} A_{yn_i n_j}^2(A_{un_i}, B_{un_i}) + B_{yn_i n_j}^2(A_{un_i}, B_{un_i}) = \mathbf{E}_{n_i}^T \cdot \mathbf{E}_{n_i} \\
 &= \left[A_{yn_i 1} \quad \dots \quad A_{yn_i n_s} \quad B_{yn_i 1} \quad \dots \quad B_{yn_i n_s} \right] \cdot \begin{bmatrix} A_{yn_i 1} \\ \vdots \\ A_{yn_i n_s} \\ B_{yn_i 1} \\ \vdots \\ B_{yn_i n_s} \end{bmatrix}
 \end{aligned} \tag{5}$$

At the minimum of $V_{n_i}(A_{un_i}, B_{un_i})$ the gradient of this function $\hat{V}_{n_i}(\hat{A}_{un_i}, \hat{B}_{un_i})$, must be zero, so that:

$$\left. \frac{\partial V_{n_i}}{\partial \theta_{n_i}} \right|_{\hat{\theta}_{n_i}} = \begin{bmatrix} \frac{\partial V_{n_i}}{\partial A_{un_i}} \\ \frac{\partial V_{n_i}}{\partial B_{un_i}} \end{bmatrix} \bigg|_{(\hat{A}_{un_i}, \hat{B}_{un_i})} = 2 \left. \frac{\partial \mathbf{E}_{n_i}}{\partial \theta_{n_i}} \right|_{\hat{\theta}_{n_i}}^T \cdot \mathbf{E}_{n_i} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{6}$$

where $\theta_{n_i} = [A_{un_i} \ B_{un_i}]^T$ and $\frac{\partial \mathbf{E}_{n_i}}{\partial \theta_{n_i}}$ is the Jacobian matrix ($\mathbf{J}$) of the column vector

$\mathbf{E}_{n_i}$ with respect to the parameter vector (θ_{n_i}) related with harmonic n_i . This parameter vector consists of the real and imaginary parts of the n_i harmonic of the command signal. $\mathbf{J}$ is nothing but the matrix with all the sensitivities and can be calculated as:

$$\mathbf{J} = \frac{\partial \mathbf{E}_{n_i}}{\partial \theta_{n_i}} = \begin{bmatrix} \frac{\partial A_{yn_i 1}}{\partial A_{un_i}} & \frac{\partial A_{yn_i 1}}{\partial B_{un_i}} \\ \vdots & \vdots \\ \frac{\partial A_{yn_i n_s}}{\partial A_{un_i}} & \frac{\partial A_{yn_i n_s}}{\partial B_{un_i}} \\ \frac{\partial B_{yn_i 1}}{\partial A_{un_i}} & \frac{\partial B_{yn_i 1}}{\partial B_{un_i}} \\ \vdots & \vdots \\ \frac{\partial B_{yn_i n_s}}{\partial A_{un_i}} & \frac{\partial B_{yn_i n_s}}{\partial B_{un_i}} \end{bmatrix} \quad (7)$$

If perturbations are small, the gradient function can be approximated using the first-order term of the Taylor Series calculated around the present point:

$$\left. \frac{\partial V_{n_i}}{\partial \theta_{n_i}} \right|_{\hat{\theta}_{n_i}} \approx \left. \frac{\partial V_{n_i}}{\partial \theta_{n_i}} \right|_{\theta_{n_i}} + \left. \frac{\partial^2 V_{n_i}}{\partial \theta_{n_i}^2} \right|_{\theta_{n_i}} (\hat{\theta}_{n_i} - \theta_{n_i}) = 2 \left. \frac{\partial \mathbf{E}_{n_i}}{\partial \theta_{n_i}} \right|_{\theta_{n_i}}^T \cdot \mathbf{E}_{n_i} + 2 \left. \frac{\partial \mathbf{E}_{n_i}}{\partial \theta_{n_i}} \right|_{\theta_{n_i}}^T \cdot \left. \frac{\partial \mathbf{E}_{n_i}}{\partial \theta_{n_i}} \right|_{\theta_{n_i}} (\hat{\theta}_{n_i} - \theta_{n_i}) \approx \begin{bmatrix} 0 \\ 0 \end{bmatrix} \quad (8)$$

where second derivatives have been neglected when calculating the Hessian matrix of the cost function (Gauss-Newton method). The necessary changes in the parameter vector to reach the cost-function optimum can be calculated as:

$$\begin{aligned} \hat{\theta}_{n_i} - \theta_{n_i} &= \begin{bmatrix} \hat{A}_{un_i} - A_{un_i} \\ \hat{B}_{un_i} - B_{un_i} \end{bmatrix} = - \left. \frac{\partial^2 V_{n_i}}{\partial \theta_{n_i}^2} \right|_{\theta_{n_i}}^{-1} \left. \frac{\partial V_{n_i}}{\partial \theta_{n_i}} \right|_{\theta_{n_i}} = - \left(\left. \frac{\partial \mathbf{E}_{n_i}}{\partial \theta_{n_i}} \right|_{\theta_{n_i}}^T \cdot \left. \frac{\partial \mathbf{E}_{n_i}}{\partial \theta_{n_i}} \right|_{\theta_{n_i}} \right)^{-1} \left. \frac{\partial \mathbf{E}_{n_i}}{\partial \theta_{n_i}} \right|_{\theta_{n_i}}^T \cdot \mathbf{E}_{n_i} = \\ &= -(\mathbf{J}_{n_i}^T \mathbf{J}_{n_i})^{-1} \mathbf{J}_{n_i}^T \cdot \mathbf{E}_{n_i} \end{aligned} \quad (9)$$

If the problem were linear, (9) would directly give the necessary command signal to eliminate the compressor vibration in a single step. However, the calculation is based on a cause-effect model which is only an approximation. Therefore, it is necessary to limit the command-signal changes in the direction established by the algorithm. This limit has to be defined by the user and it can be a different value for real and imaginary parts of each harmonic. The magnitude of these limits will influence the speed with which the algorithm progresses towards the optimum value of the cost function: while large limits will make a fast progress possible, they will also give more problems when unmodeled effects are important. Besides, a large limit may also case inconveniently large and

sudden changes in the vibration pattern of the compressor. The actual command signal is calculated summing the value applied during the previous sampling interval and the increment calculated with the sensitivities. Using complex notation, it yields, for each harmonic (n_i):

$$A_{un_i}[k] + jB_{un_i}[k] = \Delta A_{un_i}[k] + j\Delta B_{un_i}[k] + [A_{un_i}[k-1] + jB_{un_i}[k-1]] \quad (10)$$

Only those harmonics above the established limits should be perturbed at each iteration, and the process will go on as long as the “total” amplitude of, at least, one of the harmonic components in the vibration is above a predefined limit. The “total” amplitude of each harmonic M_{n_i} (n_i is the harmonic number) is calculated taking all accelerometers into account as:

$$M_{n_i} = \left(\frac{1}{n_s} \right) \sum_{n_j=1}^{n_s} \sqrt{A_{yn_i n_j}^2 + B_{yn_i n_j}^2} \quad (11)$$

The limit value for each M_{n_i} can be set independently by the user.

3.3 Command-signal generation

The periodic command signal to be followed by the slave motor (derived from the master command signal and the vibration-cancelation algorithm) is written for each harmonic in (10) as a complex number. The actual application of this command requires its translation into a real-world discrete signal ($U_{n_i \text{COMMAND}}$). This can be done, for each harmonic (n_i), with a state variable model such as:

$$\begin{bmatrix} X_{n_i \cos}[k+1] \\ X_{n_i \sin}[k+1] \end{bmatrix} = \begin{bmatrix} \cos\left(\frac{2\pi f_o n_i}{f_s}\right) & -\sin\left(\frac{2\pi f_o n_i}{f_s}\right) \\ \sin\left(\frac{2\pi f_o n_i}{f_s}\right) & \cos\left(\frac{2\pi f_o n_i}{f_s}\right) \end{bmatrix} \begin{bmatrix} X_{n_i \cos}[k] \\ X_{n_i \sin}[k] \end{bmatrix} \quad (12)$$

$$U_{n_i \text{COMMAND}}[k] = \begin{bmatrix} A_{un_i} & -B_{un_i} \end{bmatrix} \begin{bmatrix} X_{n_i \cos}[k] \\ X_{n_i \sin}[k] \end{bmatrix}$$

Where f_o is the fundamental frequency and f_s is the “fast” sampling frequency. All harmonics are then summed, yielding:

$$U_{\text{COMMAND}}[k] = \sum_{n_i} U_{n_i \text{COMMAND}}[k] \quad (13)$$

An overview of the vibration-cancellation algorithm, related to each harmonic, is shown in Fig. 3, using SIMULINK. An enable signal is provided in order to activate the algorithm when required.

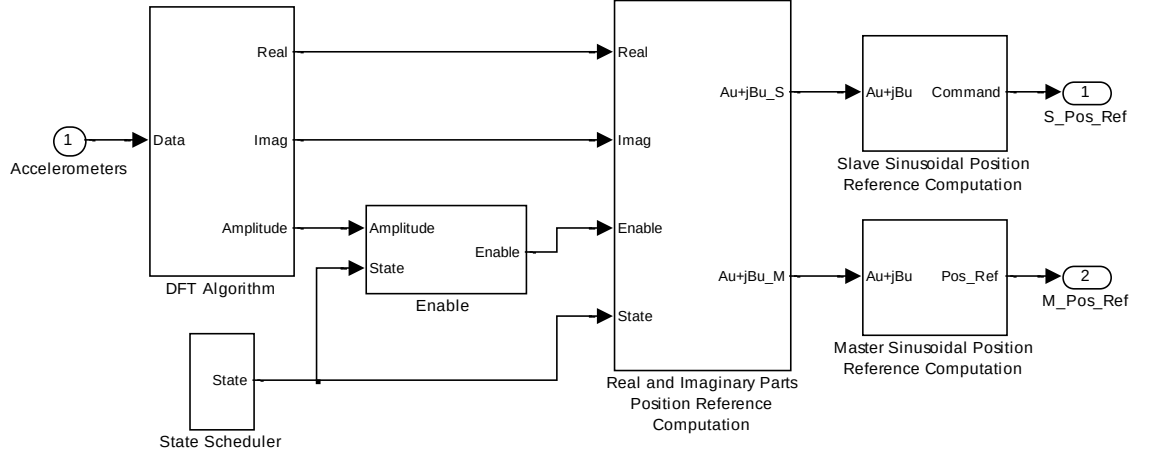


Fig. 3: Command-signal calculation in SIMULINK.

3.4 Implementation general details

3.4.1 DFT implementation

In order to avoid a large data buffer when calculating the DFT of the accelerometer measurements, a recursive implementation of the DFT with Goertzel's algorithm has been used⁵.

Calling $T[k]$ the accelerometer measurement at sampling instant k and if $T(z)$ is the Z-Transform of the input sequence, one can write, for each harmonic n_i :

$$\frac{E(z)}{T(z)} = \frac{1 - W_N^{n_i} \cdot z^{-1}}{1 - \left(2 \cdot \cos \left(\frac{2\pi \cdot n_i}{N_p} \right) \cdot z^{-1} - z^{-2} \right)} \cdot Norm \quad (14)$$

$$W_N^{n_i} = \exp \left(-j \left(\frac{2\pi \cdot n_i}{N_p} \right) \right) = \cos \left(\frac{2\pi \cdot n_i}{N_p} \right) + j \cdot \sin \left(\frac{2\pi \cdot n_i}{N_p} \right) \quad (15)$$

$$\frac{I(z)}{T(z)} = \frac{1}{1 - \left(2 \cdot \cos \left(\frac{2\pi \cdot n_i}{N_p} \right) \cdot z^{-1} - z^{-2} \right)} \quad (16)$$

⁵ See R.G. Lyons *Understanding Digital Signal Processing*, 3rd edition. Pearson Education Inc. 2011

$$\frac{A(z)}{T(z)} = \frac{1 - \cos\left(\frac{2\pi \cdot n_i}{N_p}\right) \cdot z^{-1}}{1 - \left(2 \cdot \cos\left(\frac{2\pi \cdot n_i}{N_p}\right) \cdot z^{-1} - z^{-2}\right)} = \frac{I(z)}{T(z)} (1 - K_c z^{-1}) \quad (17)$$

$$\frac{B(z)}{T(z)} = \frac{\sin\left(\frac{2\pi \cdot n_i}{N_p}\right) \cdot z^{-1}}{1 - \left(2 \cdot \cos\left(\frac{2\pi \cdot n_i}{N_p}\right) \cdot z^{-1} - z^{-2}\right)} = \frac{I(z)}{T(z)} K_s \cdot z^{-1} \quad (18)$$

where,

$$Norm = \left(\frac{2}{N_c \cdot N_p} \right)$$

N_c = number of cycles to be used in the DFT.

N_p = number of points per cycle

and $E[k]$ is the DFT result for harmonic n_i (see eq. (14)), but only after $N_c \times N_p$ samples and it is a complex number: $E[k] = A[k] + j \cdot B[k]$.

If real and imaginary parts of $E[k]$ are going to be processed separately, one can write the following difference equations:

$$I[k] = T[k] + 2K_c I[k-1] - I[k-2] \quad (19)$$

$$A[k] = (I[k] - K_c \cdot I[k-1]) \cdot Norm \quad (20)$$

$$B[k] = K_s \cdot I[k-1] \cdot Norm \quad (21)$$

with $Norm$ as in (14) and:

$$K_c = \cos\left(\frac{2\pi \cdot n_i}{N_p}\right), \quad K_s = \sin\left(\frac{2\pi \cdot n_i}{N_p}\right)$$

After $(N_c \cdot N_p)$ samples the value of $E[k]$ is latched to be used in the rest of the algorithm while all variables used in Goertzel's algorithm are reset to zero. Goertzel's algorithm is depicted in Fig. 4. The actual implementation has split the algorithm in two parts: one is carried out using fixed-point arithmetic (eq. (19) and FPGA in Fig. 4) and the other one is carried out in floating-point arithmetic (eqs. (20), (21) and Fig. 4).

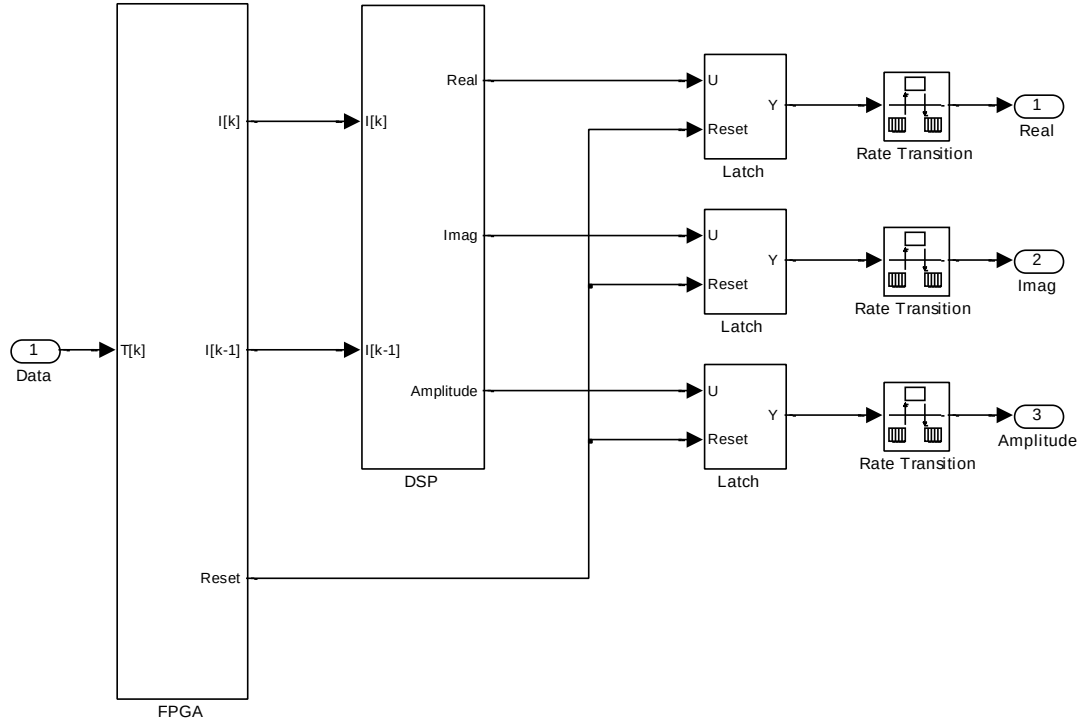


Fig. 4: Goertzel's algorithm. Overview.

Note: Fixed-point implementation of Goertzel's algorithm in CDE

Most of the implementation of Goertzel's algorithm has been done in a VHDL block inside the CDE control FPGA (the block is called *apb2dft* in the project). The fixed-point part of Goertzel's algorithm has been coded as follows:

- The relation between real-world variables (I and T) and fixed-point magnitudes (Q) is:

$$I[k] = Q_{I[k]} LSB_I$$

$$T[k] = Q_T \cdot LSB_T$$

$$LSB_I = LSB_T = \frac{5}{2.128} \frac{1}{2^{12}}$$

$$2^{Kc} = Q_{2^{Kc}} LSB_{2^{Kc}}$$

$$LSB_{2^{Kc}} = 2^{-30}$$

- The algorithm proceeds as follows:

$$Q_{I[k]} LSB_I = Q_T LSB_I + Q_{2^{Kc}} LSB_{2^{Kc}} \cdot Q_{I[k-1]} LSB_I - Q_{I[k-2]} LSB_I$$

$$Q_{I[k]} LSB_I = (Q_T + 2 \cdot Kc \cdot Q_{I[k-1]} \cdot 2^{-30} - Q_{I[k-2]}) \cdot LSB_I$$

$$Q_{I[k]} = (2^{30} Q_T + Q_{2^{Kc}} \cdot Q_{I[k-1]} - 2^{30} Q_{I[k-2]}) \cdot 2^{-30}$$

where Q_x are integers: Q_T is a 13-bit signed integer and Q_{2Kc} is a 32-bit integer. Q_I will not exceed 32 bits in any case.

Real and imaginary parts of $I[k]=Q_{I[k]}LSB_I$ (see eqs. (20) and (21)), when ready, are available for telemetry with $(1/LSB)=27777.7$.

3.4.2 Computation of the real and imaginary parts of the position reference for the slave compressor

This block calculates the command to be used in the slave motor at each iteration. This command signal contains the harmonics to be applied to the position reference of the slave motor in order to achieve the programmed objective. Three states are possible:

1. Perturbing real part of the command-signal components.
2. Perturbing imaginary part of the command-signal components.
3. New position reference is calculated (using sensitivities) and applied.

These states run selected sequentially using a state scheduler.

An overview of how the command signal is calculated is depicted in Fig. 5. The three stages presented in that figure are explained bellow. Before this explanation, the way in which equation (9) has been implemented must be spelled out.

1. Calling

- a. $dX = \begin{pmatrix} da \\ db \end{pmatrix}$ the perturbation applied to the real part (da) and the imaginary part (db).
- b. $E = \begin{pmatrix} A_e \\ B_e \end{pmatrix}$, E accelerometer amplitude without perturbation
- c. $E_{da} = \begin{pmatrix} A_{da} \\ B_{da} \end{pmatrix}$, E_{da} accelerometer amplitude with perturbation da
- d. $E_{db} = \begin{pmatrix} A_{db} \\ B_{db} \end{pmatrix}$, E_{db} accelerometer amplitude with perturbation db

2. J can be calculated as:

$$J = \begin{pmatrix} \frac{1}{da}(A_{da} - A_e) & \frac{1}{db}(A_{db} - A_e) \\ \frac{1}{da}(B_{da} - B_e) & \frac{1}{db}(B_{db} - B_e) \end{pmatrix} = \begin{pmatrix} \frac{1}{da}S_{Ada} & \frac{1}{db}S_{Adb} \\ \frac{1}{da}S_{Bda} & \frac{1}{db}S_{Bdb} \end{pmatrix}$$

$$3. J^T \cdot E = \begin{pmatrix} \frac{1}{da}(S_{Ada} \cdot A_e + S_{Bda} \cdot B_e) \\ \frac{1}{db}(S_{Adb} \cdot A_e + S_{Bdb} \cdot B_e) \end{pmatrix}$$

$$4. J^T \cdot J = \begin{pmatrix} \frac{1}{da^2}(S_{Ada}^2 + S_{Bda}^2) & \frac{1}{dadb}(S_{Ada} \cdot S_{Adb} + S_{Bda} \cdot S_{Bdb}) \\ \frac{1}{dadb}(S_{Ada} \cdot S_{Adb} + S_{Bda} \cdot S_{Bdb}) & \frac{1}{db^2}(S_{Adb}^2 + S_{Bdb}^2) \end{pmatrix}$$

$$5. \quad (J^T \cdot J)^{-1} = \frac{1}{\det(J^T \cdot J)} \begin{pmatrix} \frac{1}{db^2} (S_{Adb}^2 + S_{Bdb}^2) & -\frac{1}{dadb} (S_{Ada} \cdot S_{Adb} + S_{Bda} \cdot S_{Bdb}) \\ -\frac{1}{dadb} (S_{Ada} \cdot S_{Adb} + S_{Bda} \cdot S_{Bdb}) & \frac{1}{da^2} (S_{Ada}^2 + S_{Bda}^2) \end{pmatrix}$$

$$6. \quad \det(J^T \cdot J) = \frac{1}{da^2 db^2} [S_{Ada}^2 \cdot S_{Bdb}^2 + S_{Bda}^2 \cdot S_{Adb}^2 - 2 \cdot S_{Ada} \cdot S_{Adb} \cdot S_{Bda} \cdot S_{Bdb}]$$

$$7. \quad \det(J^T \cdot J) = \frac{1}{da^2 db^2} (S_{Ada} \cdot S_{Bdb} - S_{Bda} \cdot S_{Adb})^2$$

$$8. \quad \det(J^T \cdot J) = \frac{1}{da^2 db^2} \Omega^2$$

9. From point 8 above, if $da=db$,

$$\det(J^T \cdot J) = \frac{1}{da^4} \cdot \Omega^2$$

$$10. \quad J^T \cdot E = \begin{pmatrix} \frac{1}{da} (S_{Ada} \cdot A_e + S_{Bda} \cdot B_e) \\ \frac{1}{db} (S_{Adb} \cdot A_e + S_{Bdb} \cdot B_e) \end{pmatrix}$$

$$11. \quad (J^T \cdot J)^{-1} \cdot J^T \cdot E = \frac{1}{\det(J^T \cdot J)} \begin{pmatrix} \frac{1}{dadb^2} [A_e \cdot S_{Bdb} \cdot \Omega - B_e \cdot S_{Adb} \cdot \Omega] \\ \frac{1}{da^2 db} [B_e \cdot S_{Ada} \cdot \Omega - A_e \cdot S_{Bda} \cdot \Omega] \end{pmatrix}$$

$$12. \quad (J^T \cdot J)^{-1} \cdot J^T \cdot E = \frac{1}{\det(J^T \cdot J)} \begin{pmatrix} \frac{1}{dadb^2} [A_e \cdot S_{Bdb} - B_e \cdot S_{Adb}] \cdot \Omega \\ \frac{1}{da^2 db} [B_e \cdot S_{Ada} - A_e \cdot S_{Bda}] \cdot \Omega \end{pmatrix}$$

$$Inc = -(J^T \cdot J)^{-1} \cdot J^T \cdot E = \frac{1}{\det(J^T \cdot J)} \begin{pmatrix} -\frac{1}{dadb^2} [A_e \cdot S_{Bdb} - B_e \cdot S_{Adb}] \cdot \Omega \\ -\frac{1}{da^2 db} [B_e \cdot S_{Ada} - A_e \cdot S_{Bda}] \cdot \Omega \end{pmatrix} =$$

$$13. \quad = \frac{1}{\det(J^T \cdot J)} \begin{pmatrix} \left[\frac{1}{dadb^2} B_e \cdot S_{Adb} - \frac{1}{dadb^2} A_e \cdot S_{Bdb} \right] \cdot \Omega \\ \left[\frac{1}{da^2 db} A_e \cdot S_{Bda} - \frac{1}{da^2 db} B_e \cdot S_{Ada} \right] \cdot \Omega \end{pmatrix}$$

$$14. \quad Inc = \frac{1}{\det(J^T \cdot J)} \begin{pmatrix} Re \quad al \\ Im \quad ag \end{pmatrix}$$

It is worth pointing out that,

$$\begin{aligned}
S_{Ada} &= A_{da} - A_e & S_{Bda} &= B_{da} - B_e \\
S_{Adb} &= A_{db} - A_e & S_{Bdb} &= B_{db} - B_e \\
\Omega &= S_{Ada} \cdot S_{Bdb} - S_{Bda} \cdot S_{Adb} \\
\Omega &= D_{AdaBdb} - D_{BdaAdb} \\
\text{Re } al &= \left[\frac{1}{da^3} B_e \cdot S_{Adb} - \frac{1}{da^3} A_e \cdot S_{Bdb} \right] \cdot \Omega = [C_{Adb} - C_{Bdb}] \cdot \Omega = \text{Re } al_{int} \cdot \Omega \\
\text{Im } ag &= \left[\frac{1}{da^3} A_e \cdot S_{Bda} - \frac{1}{da^3} B_e \cdot S_{Ada} \right] \cdot \Omega = [C_{Bda} - C_{Ada}] \cdot \Omega = \text{Im } ag_{int} \cdot \Omega \\
\begin{bmatrix} \text{Inc}_{real} \\ \text{Inc}_{imag} \end{bmatrix} &= \begin{bmatrix} \frac{\text{Re } al_{int} \cdot \Omega}{da^4} \\ \frac{\text{Im } ag_{int} \cdot \Omega}{da^4} \end{bmatrix}
\end{aligned}$$

3.4.3 Perturbing real part of the command-signal components

This stage is represented by box “stage1” in Fig.5.

Stage 1

- Inputs of stage1 :
 - N_c (parameter) Number of cycles to be used in the FFT
 - N_p (parameter) Number of points per cycle
 - $I[k]$ and $I[k-1]$ for all harmonics considered
 - da (see above) and $1_{da3}=1/da^3$
 - v_{amp_sat} : amplitude saturation
 - v_m : Control threshold
 - $v_{amp_dem_vibr}$ and $v_{pha_dem_vibr}$: command signal without perturbation (amplitude and phase)
- Outputs :
 - A_e and B_e for harmonics of interest
 - $1_{da3}A_e=(1/da^3)A_e$ and $1_{da3}B_e=(1/da^3)B_e$ for harmonics of interest
 - v_{amp} : amplitude of DFT
 - part v_{real_act} and v_{imag_act} : command signal without perturbation in real and imaginary part
 - $v_{amp_dem_vibr}$ and $v_{pha_dem_vibr}$: new command signal with perturbation in real part (amplitude and phase)

For the actual implementation,

- $da=Q_{da} LSB_{da}$
- For fundamental harmonic $LSB_{da}=6.1037 \times 10^{-6}$
- For other harmonics $LSB_{da}=6.1037 \times 10^{-6}$
- $1/da^3=Q_{da^3} \cdot LSB_{da^3}$
- For the fundamental harmonic $LSB_{da^3} = 8 \times 10^3$
- For other harmonics $LSB_{da^3} = 1.01 \times 10^6$

- $I[k] = Q_{I[k]} LSB_I$ $LSB_I = \frac{5}{2.128 \times 2^{12}}$
- $M = Q_M LSB_M$, Amplitud limit (one for each harmonic) $LSB_M = 0.0000061035$
- $V_amp_sat = Q_{amp_sat} LSB_{amp_sat}$ and $LSB_{amp_sat} = 0.000305175$

Where Q_x are 32-bit integers and LSB and 2π are coded as 24-bits floating point numbers.

- A_e and B_e (Goertzel's algorithm), $(1/da3)A_e$ and $(1/da3)B_e$ are calculated within this stage. Furthermore, some calculations are performed in the actual implementation
- $A_e = Norm * (I_n - Kc . I_{n-1})$
- $B_e = Norm * Ks . I_{n-1}$
- $v_real_act = v_amp_dem_vibr \cdot \cos(v_pha_dem_vibr)$
 $v_imag_act = v_amp_dem_vibr \cdot \sin(v_pha_dem_vibr)$
 $v_amp_dem_vibr = \sqrt{(v_real_act + da)^2 + (v_imag_act)^2}$
 $v_pha_dem_vibr = \text{phase}(v_real_act + da, v_imag_act)$

The amplitude ($v_amp_dem_vibr$) of the command signal is saturated to v_amp_sat . The *phase* function is calculated with a simplified implementation of the *arctan* function (see note at the end of this Section).

The calculated command signal perturbing real part is applied and the amplitude of the vibration is measured as

$$v_amp = A_e^2 + B_e^2$$

to evaluate if the algorithm has to go into stage2. The square root of v_amp is also calculated here and compared with the limit M .

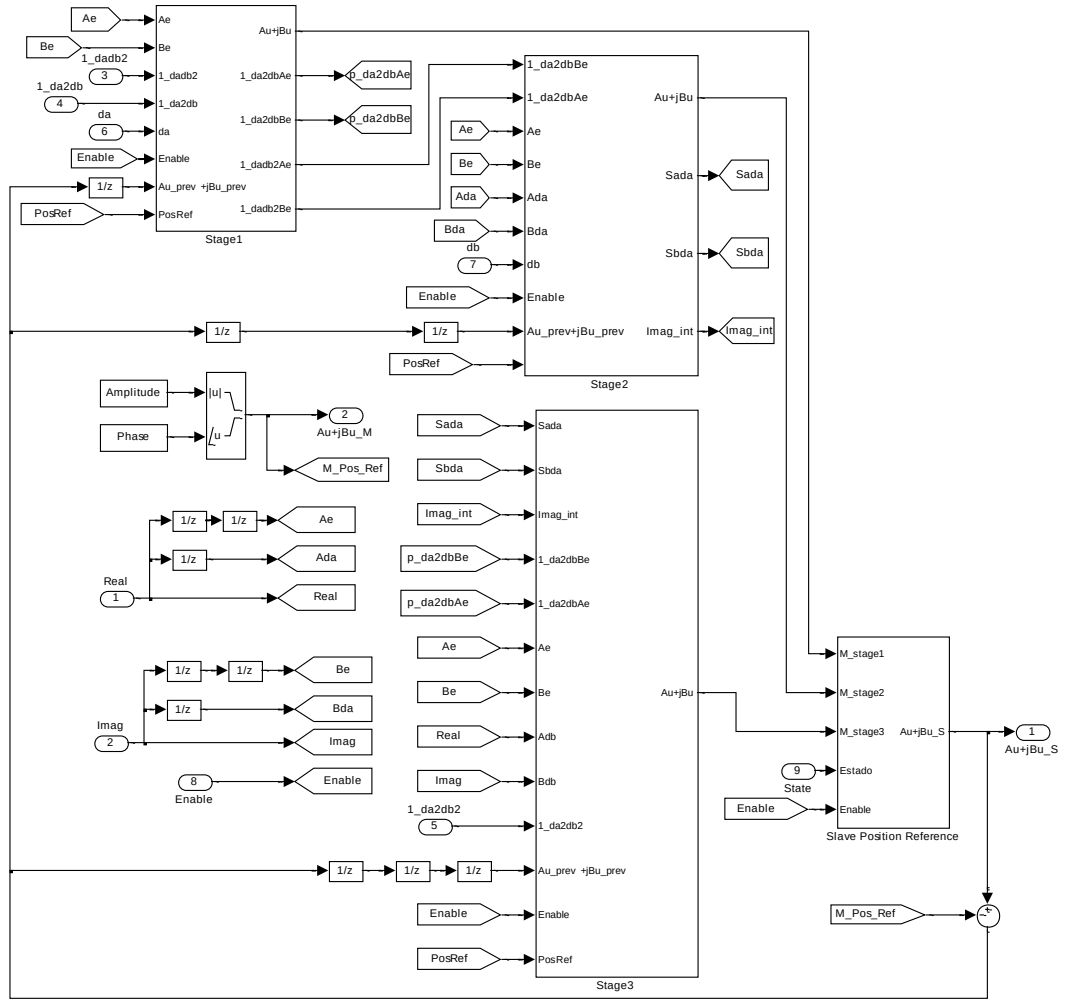


Fig. 5: Block diagram describing the three stages of the algorithm.

3.4.4 Perturbing imaginary part of the command-signal components

This stage is represented by box “stage2” in Fig.5.

Stage2

- Inputs :
 - N_c (parameter) is the number of cycles in the DFT
 - N_p (parameter) is the number of points per cycle in the DFT
 - $I[k]$ and $I[k-1]$ for all harmonics considered
 - da (see above)
 - v_{amp_sat} amplitude saturation
 - A_e and B_e for harmonics
 - $1_{da3Ae} = A_e \cdot 1/(da^3)$ and $1_{da3Be} = B_e \cdot 1/(da^3)$ for harmonics
 - Command signal without perturbation (real and imaginary parts) v_{real_act} and v_{imag_act}
- Outputs :

- S_{Ada} and S_{Bda} for harmonics
- V_{imag_int} for harmonics
- New command signal with perturbation in imaginary part
 $v_{amp_dem_vibr}$ and $v_{pha_dem_vibr}$

Notice that,

- $da = Q_{da} LSB_{da}$
- For fundamental harmonic $LSB_{da} = 6.1037 \times 10^{-6}$
- For other harmonics $LSB_{da} = 6.1037 \times 10^{-6}$
- $I[k] = Q_{I[k]} LSB_I$ $LSB_I = \frac{5}{2.128 \times 2^{12}}$
- $V_{amp_sat} = Q_{amp_sat} LSB_{amp_sat}$ and $LSB_{amp_sat} = 0.000305175$
- Where Q_x are 32-bit integers and LSB and 2π are coded in 24-bits floating point numbers
- Further calculation include:
 - $Ada = Norm \cdot (I_n - Kc \cdot I_{n-1})$
 - $Bda = Norm \cdot Ks \cdot I_{n-1}$
 - $S_{Ada} = A_{da} - A_e$
 - $S_{Bda} = B_{da} - B_e$
 - $C_{Ada} = B_e \cdot S_{Ada} \cdot 1/(da^3)$
 - $C_{Bda} = A_e \cdot S_{Bda} \cdot 1/(da^3)$
 - $Im_ag_int = C_{Bda} - C_{Ada}$
 - $v_{amp_dem_vibr} = \frac{\sqrt{(v_{real_act})^2 + (v_{imag_act} + da)^2}}{da}$
 - $v_{pha_dem_vibr} = phase(v_{real_act}, v_{imag_act} + da)$

3.4.5 Updating the command-signal for the slave compressor

Stage3

- Inputs :
 - N_c (parameter) is the number of cycles in the DFT
 - N_p (parameter) is the number of points per cycle in the DFT
 - $I[k]$ and $I[k-1]$ for all harmonics considered
 - $da4 = da^4$ (see above)
 - v_{amp_sat} amplitude saturation
 - v_{inc_sat} Incremental amplitude saturation
 - A_e and B_e for harmonics
 - $1_da3Ae = A_e \cdot 1/(da^3)$ and $1_da3Be = B_e \cdot 1/(da^3)$ for harmonics
 - S_{Ada} and S_{Bda} for harmonics
 - V_{imag_int} for harmonics
 - Command signal without perturbation (real and imaginary parts)
 v_{real_act} and v_{imag_act}
- Outputs :
 - New command signal with the increments calculated to reduce vibration

For the actual implementation:

- $da = Q_{da} LSB_{da}$
- For fundamental harmonic $LSB_{da} = 6.1037 \times 10^{-6}$
- For other harmonics $LSB_{da} = 6.1037 \times 10^{-6}$
- $da^4 = Q_{da^4} \cdot LSB_{da^4}$
- $LSB_{da^4} = 2.05 \times 10^{-10}$ fundamental harmonic
- $LSB_{da^4} = 37.225 \times 10^{-14}$ harmonics

- $I_n = Q_{I_n} LSB_I \quad LSB_I = \frac{5}{2.128 \times 2^{12}}$
- $V_{amp_sat} = Q_{amp_sat} LSB_{amp_sat}$ and $LSB_{amp_sat} = 0.000305175$
- $v_{inc_sat} = Q_{inc_sat} LSB_{inc_sat}$ and $LSB_{inc_sat} = 0.00013733$
- Where Q_x are 32-bit integers and LSB and 2π are coded as 24-bits floating point numbers.
- Further calculation include:
 - $A_{db} = Norm \cdot (I_n - Kc \cdot I_{n-1})$
 - $B_{db} = Norm \cdot Ks \cdot I_{n-1}$
 - $S_{Adb} = A_{db} - A_e$
 - $S_{Bdb} = B_{db} - B_e$
 - $C_{Bdb} = A_e \cdot S_{Bdb} \cdot (1 / da^3)$
 - $C_{Adb} = B_e \cdot S_{Adb} \cdot (1 / da^3)$
 - $D_{BdaAdb} = S_{Bda} \cdot S_{Adb}$
 - $D_{AdaBdb} = S_{Bdb} \cdot S_{Ada}$
 - $Re_al_int = C_{Bdb} - C_{Adb}$
 - $Inc_Re_al = (1 / \Omega) \cdot Re_al_int \cdot da^4$
 - $Inc_Im_ag = (1 / \Omega) \cdot Im_ag_int \cdot da^4$
 - $v_{amp_inc} = \sqrt{Inc_real^2 + Inc_imag^2}$ with saturation v_{inc_sat}
 - $v_pha_inc = \arctan(Inc_real / Inc_imag)$
 - $Inc_real = v_amp_inc \cdot \cos(v_pha_inc)$
 - $Inc_imag = v_amp_inc \cdot \sin(v_pha_inc)$
 - $v_{amp_dem_vibr} = \frac{\sqrt{(v_real_act + Inc_real)^2 + (v_imag_act + Inc_imag)^2}}{2}$
 - $v_pha_dem_vibr = phase(v_real_act + Inc_real, v_imag_act + Inc_imag)$

At the beginning of each state, the appropriate command signal is applied according to the state scheduler. The latency cycles are observed before measurements can be taken. Unlike in the theoretical explanation of Section 3.4, in the implementation the command signal is generated from the values of amplitude and phase in every stage.

Note: Proposed method to calculated the phase of a complex number (taken from R.G. Lyons Understanding Digital Signal Processing, 3rd edition. Pearson Education Inc. 2011)

Given a complex number $a = I + jQ$

- The 360° are divided into 8 sectors
- The sector in which the complex number is, can be calculated as follows:

Sign(I)	Sign(Q)	$ Q - I $	Sector	Phase
+	+	$ Q - I < 0$	1°	0-45°
+	+	$ Q - I \geq 0$	2°	45°-90°
-	+	$ Q - I \geq 0$	3°	90°-135°
-	+	$ Q - I < 0$	4°	135°-180°
-	-	$ Q - I < 0$	5°	180°-225°
-	-	$ Q - I \geq 0$	6°	225°-270°
+	-	$ Q - I \geq 0$	7°	270°-315°
+	-	$ Q - I < 0$	8°	315°-360°

The phase of the complex number is calculated as

$$\alpha \text{ (rad)} = \frac{IQ}{I^2 + 0.28125Q^2} \text{ 1st and 8th}$$

$$\alpha = \left(\frac{\pi}{2}\right) - \frac{IQ}{Q^2 + 0.28125I^2} \text{ 2nd and 3rd}$$

$$\alpha = \text{sign}(Q)\pi + \frac{IQ}{I^2 + 0.28125Q^2} \text{ 4th and 5th}$$

$$\alpha = -\left(\frac{\pi}{2}\right) - \frac{IQ}{Q^2 + 0.28125I^2} \text{ 6th and 7th}$$

The product $0.28125X^2$ can be more easily calculated as

$$0.28125X^2 = \left(\frac{1}{4} + \frac{1}{32}\right)X^2$$

The maximum error is 0.26 degrees

Note: Vibration-cancellation algorithm using the concept of transfer function

The increments to be applied to the harmonic components of the command signal in the slave motor can also be calculated inverting the transfer function which relates command and vibration. Since the problem is restricted to a number of frequencies, the frequency response of that transfer function has to be used. In theory, one should calculate as many transfer functions as accelerometers one has. However, since only one command signal is available, the command to be applied will be calculated as the average of the command values obtained for each accelerometer⁶. This procedure cannot be recommended if the accelerometer measurements were not redundant, somehow. This limitation does not apply to the procedure described in Section 3.2, since an optimization problem is solved in that case.

The estimation of the frequency response of the transfer function for harmonic n_i and accelerometer n_j is also carried out using the sensitivity information presented before:

$$F_{A_{n_i n_j}} + jF_{B_{n_i n_j}} = \frac{1}{2} \left[\left(\frac{\partial A_{yn_i n_j}}{\partial A_{un_i}} + j \frac{\partial B_{yn_i n_j}}{\partial A_{un_i}} \right) + \left(\frac{\partial B_{yn_i n_j}}{\partial B_{un_i}} - j \frac{\partial A_{yn_i n_j}}{\partial B_{un_i}} \right) \right] \quad (22)$$

The command change in harmonic n_i of the command signal is calculated as:

$$\Delta A_{un_i} + j\Delta B_{un_i} = - \frac{1}{n_s} \sum_{k=1}^{n_s} \frac{A_{yn_i n_j} + jB_{yn_i n_j}}{F_{B_{n_i n_j}} + jF_{A_{n_i n_j}}} \quad (23)$$

Summarizing, one calculates the necessary command to cancel each harmonic present in the vibration signal inverting the estimating transfer function and averaging the contribution of each command-accelerometer pair.

This alternative has many other parts (data preparation and harmonic analysis) which are not described in this section because they are common to the algorithm described in Section 3.2.

⁶ The procedure described here is an extension for several accelerometers of the patent Shaun L, Champlon, Yeong-Wei A. Wu, Michael H. Kieffer. "Adaptive feed-forward vibration control system and method". *US Patent 5,836,165*. Nov. 17, 1998. Besides, in this patent, the transfer function is known while here it is estimated from the sensitivity information.

3.5 Parameters of the vibration-cancellation algorithm

The following parameters must be set by the user:

1. Fundamental frequency
2. Number of accelerometers placed on the device
3. Number of samples for the Goertzel's algorithm (equivalent to length of the FFT window)
4. Waiting time after a command change takes place before new measurements can be taken
5. Set increment value for the real and imaginary part of the command to obtain the sensibilities (for the first harmonic).
6. Set increment value for the real and imaginary part of the command to obtain the sensibilities (for all harmonics but the first one).
7. Maximum increment to be applied when updating the harmonic amplitude in the command (for the first harmonic).
8. Maximum increment to be applied when updating the harmonic amplitude in the command (for all harmonics but the first one).
9. Vibration amplitude target for each harmonic.
10. Vibration control output saturation value for the first harmonic.
11. Vibration control output saturation value for all harmonics but the first one

Goertzel's algorithm is used to compute the complex amplitude of each harmonic. This is a recursive form of a DFT computed at individual frequencies. The number of samples needed to compute the complex amplitude in Goertzel's algorithm is equivalent to the length of the DFT, but does not need to be a power of 2 as in the case of a FFT. When the number of individual frequencies to measure is smaller than $\log_2(N)$, where N is the number of samples (or length of the DFT), the number of operations is smaller with the Goertzel's algorithm than with a typical FFT. Another important advantage in the present application is the fact that the algorithm is basically a 2nd order IIR filter, so that only 2 registers are needed to operate, in contrast to the large amount of memory needed for the buffer if a FFT were used.

Regarding the selection of cooler working frequency, due to the nature of the algorithm, bin leakage and slide of the phase between windows can occur if the sampling frequency is not a multiple of the working frequency. These problems have the same effect in the Goertzel's algorithm as in the FFT. Bin leakage generates wrong amplitude measurements, while phase sliding creates a phase shift between consecutive data samples, which can mislead the sensitivity calculation, as the ratio between real and imaginary parts can change not only due to the excitation signal used by the vibration-cancellation algorithm, but also because of this phase shifting. As a result, the programmable cooler working frequency must always be divisor of the fast sampling frequency. In this project, data sampling will occur at 12.5 kHz, which restricts the usable frequencies to the values presented in Table 1.

Table 1. Possible compressor working frequencies and the equivalent frequency resolution

N (samples)	Cooler Working Frequency	dF	N (samples)	Cooler Working Frequency	dF
313	39.94 Hz	-	286	43.71 Hz	0.15 Hz
312	40.06 Hz	0.13 Hz	285	43.86 Hz	0.15 Hz
311	40.19 Hz	0.13 Hz	284	44.01 Hz	0.15 Hz
310	40.32 Hz	0.13 Hz	283	44.17 Hz	0.16 Hz
309	40.45 Hz	0.13 Hz	282	44.33 Hz	0.16 Hz
308	40.58 Hz	0.13 Hz	281	44.48 Hz	0.16 Hz
307	40.72 Hz	0.13 Hz	280	44.64 Hz	0.16 Hz
306	40.85 Hz	0.13 Hz	279	44.80 Hz	0.16 Hz
305	40.98 Hz	0.13 Hz	278	44.96 Hz	0.16 Hz
304	41.12 Hz	0.13 Hz	277	45.13 Hz	0.16 Hz
303	41.25 Hz	0.14 Hz	276	45.29 Hz	0.16 Hz
302	41.39 Hz	0.14 Hz	275	45.45 Hz	0.16 Hz
301	41.53 Hz	0.14 Hz	274	45.62 Hz	0.17 Hz
300	41.67 Hz	0.14 Hz	273	45.79 Hz	0.17 Hz
299	41.81 Hz	0.14 Hz	272	45.96 Hz	0.17 Hz
298	41.95 Hz	0.14 Hz	271	46.13 Hz	0.17 Hz
297	42.09 Hz	0.14 Hz	270	46.30 Hz	0.17 Hz
296	42.23 Hz	0.14 Hz	269	46.47 Hz	0.17 Hz
295	42.37 Hz	0.14 Hz	268	46.64 Hz	0.17 Hz
294	42.52 Hz	0.14 Hz	267	46.82 Hz	0.17 Hz
293	42.66 Hz	0.15 Hz	266	46.99 Hz	0.18 Hz
292	42.81 Hz	0.15 Hz	265	47.17 Hz	0.18 Hz
291	42.96 Hz	0.15 Hz	264	47.35 Hz	0.18 Hz
290	43.10 Hz	0.15 Hz	263	47.53 Hz	0.18 Hz
289	43.25 Hz	0.15 Hz	262	47.71 Hz	0.18 Hz
288	43.40 Hz	0.15 Hz	261	47.89 Hz	0.18 Hz
287	43.55 Hz	0.15 Hz	260	48.08 Hz	0.18 Hz

3.6 Algorithm flow for vibration cancellation

A flow chart of the vibration algorithm can be seen in Figure 6 and can be summarized as follows:

1. To start with, default complex amplitudes on all harmonics are set
2. Both compressors are ramped up to steady state
3. Once steady state is reached ...
4. Vibration is sampled
5. The DFT is applied
6. If the vibration amplitude of a harmonic ($n_i=1$ to $n_i=7$) is below the limit established, this harmonic is removed from the list of those to be cancelled. Otherwise, this harmonic is added to the list
7. If at least one harmonic is left in the “active” list the algorithm has to continue in 8. If no harmonic is present in the list the algorithm jumps back to 4
8. The real part of the harmonics in the “active” list is incremented in the command signal
9. The system waits for steady state
10. Vibration is sampled
11. The imaginary part of the harmonics in the “active” list is incremented in the command signal
12. The system waits for steady state
13. The DFT is applied
14. The sensitivities of the “active” harmonics with respects to the inputs are calculated
15. The search direction is computed from the sensibilities and the command is updated
16. Repeat from step 4

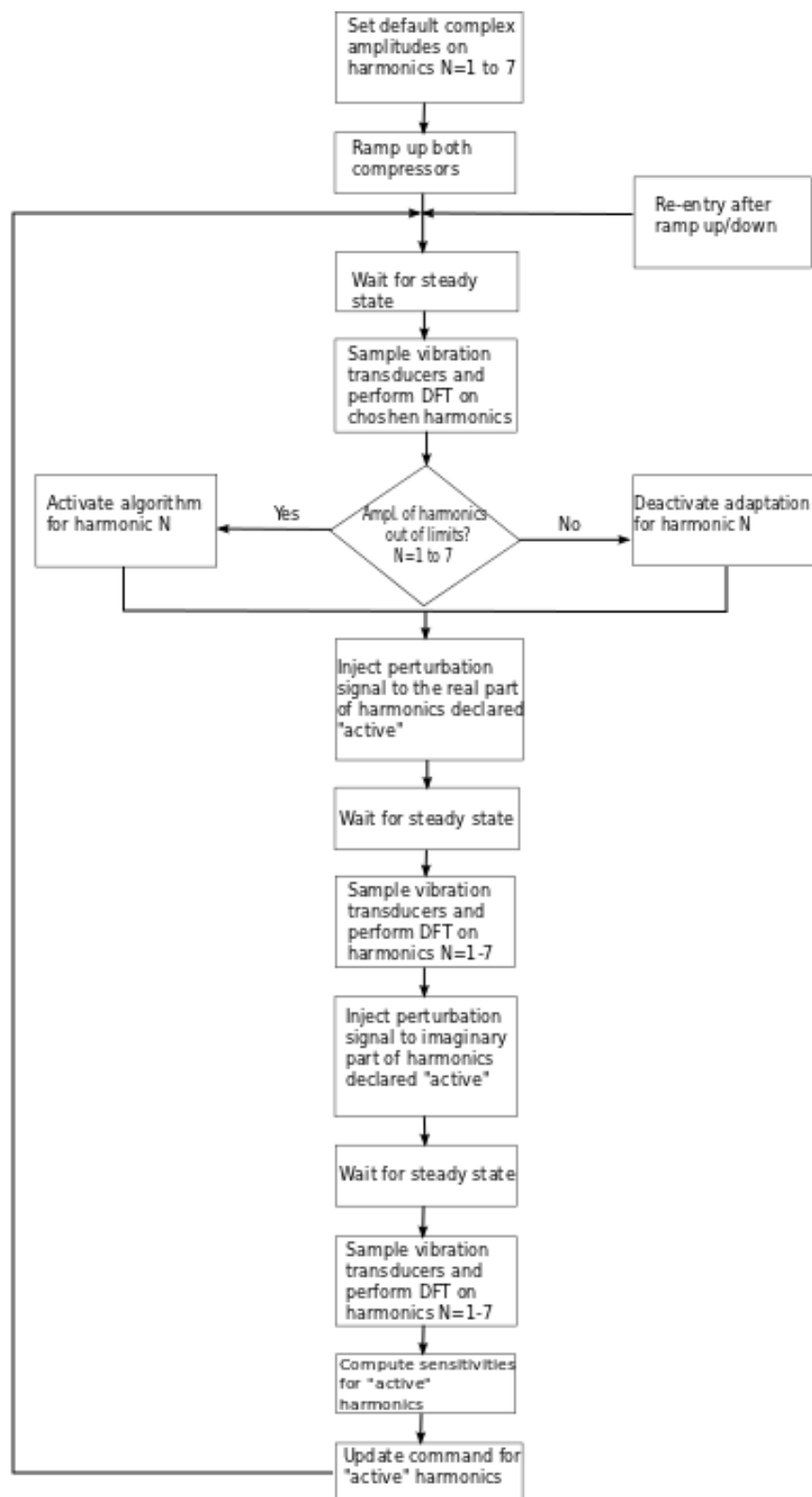


Fig. 6. Vibration-cancellation Algorithm

4. Description of the test-rig built by CRISA

The test-rig built to test the vibration-cancellation algorithm is shown in Figs. 7 and 8. The schematic of the test-rig is shown in Fig. 9. The device to be tested in the test-rig consists of two identical thermo-dynamical circuits with a compressor and a displacer each circuit. One pair compressor-displacer will work as the “master” (M Comp and M Disp are master compressor and displacer, respectively, in Fig. 9) and the other one will work as the “slave” (S Comp and S Disp are slave compressor and displacer, respectively, in Fig. 9). The master compressor and the master displacer are commanded with a sinusoidal position reference with a frequency of 43 Hz. These two sinusoidal reference signals have a programmed relative phase shift. The amplitude and the relative phase shift of these signals will affect the temperature of the so-called “cold-tip”. The slave compressor is driven by a reference signal consisting of (a) a sinusoidal component of 43 Hz, which is closely related to the one used for the master pair, and (b) the contribution of the vibration-cancellation algorithm. Finally, the slave displacer is commanded by a sinusoidal reference signal of the fundamental frequency of the slave compressor but with a phase shift.

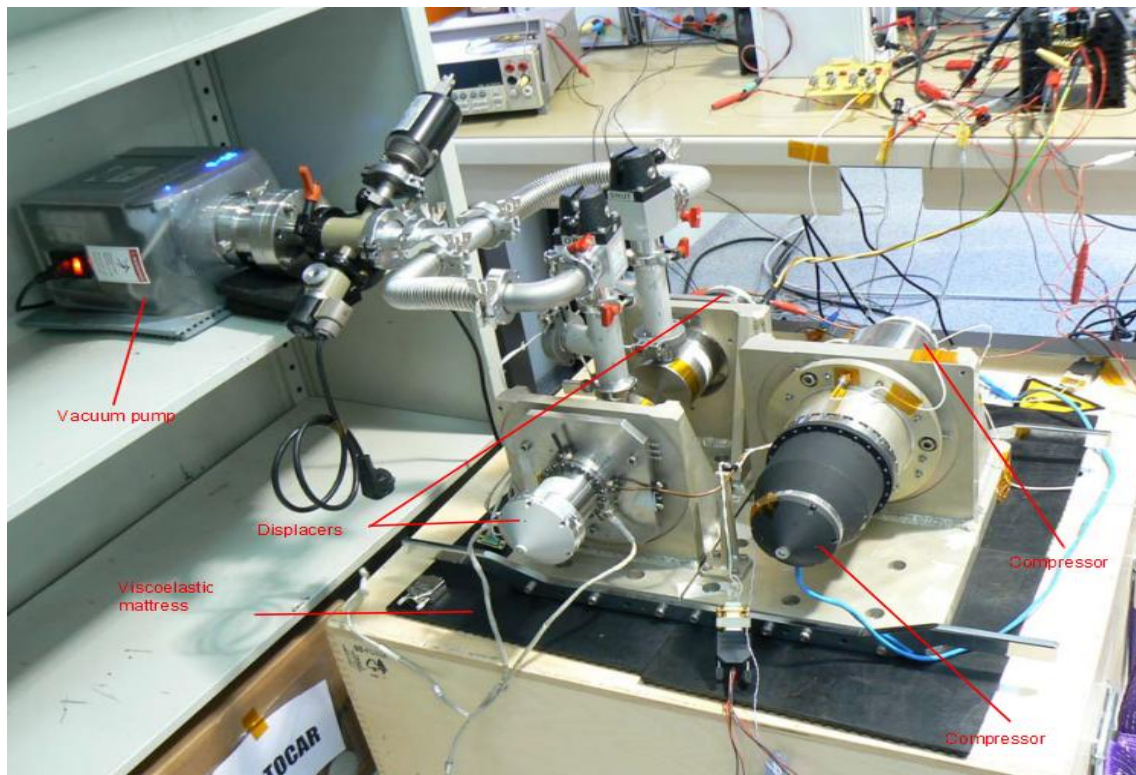


Fig. 7: Test-rig for a cryo-cooler built by CRISA

The relative positions of the two compressors (see Fig. 8) make it possible to cancel, almost completely, the vibration caused by the master compressor providing a symmetrical movement with the slave compressor. The remaining vibration caused by small asymmetries in the assembly has to be tackled by the vibration-cancellation algorithm. The vibrations in the test-rig are measured using a single accelerometer

placed as shown in Fig. 8. The approach described implies that the vibrations caused by the displacers are negligible.

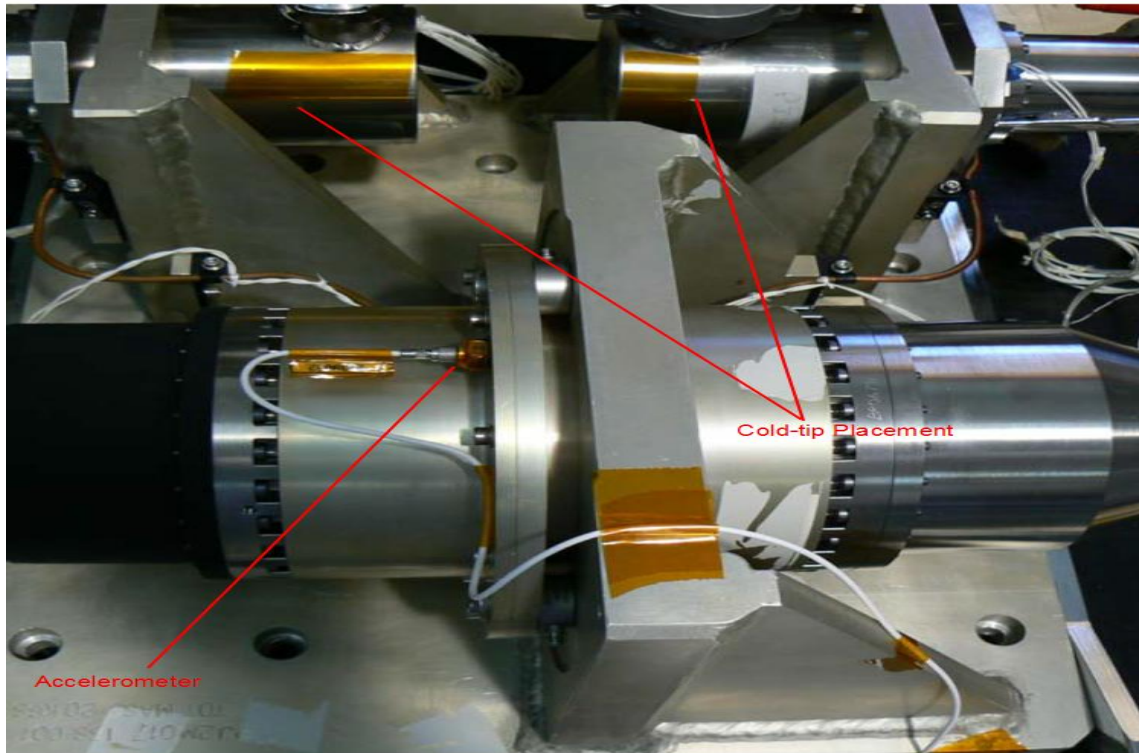


Fig. 8: Accelerometer placement to measure vibration in the test rig

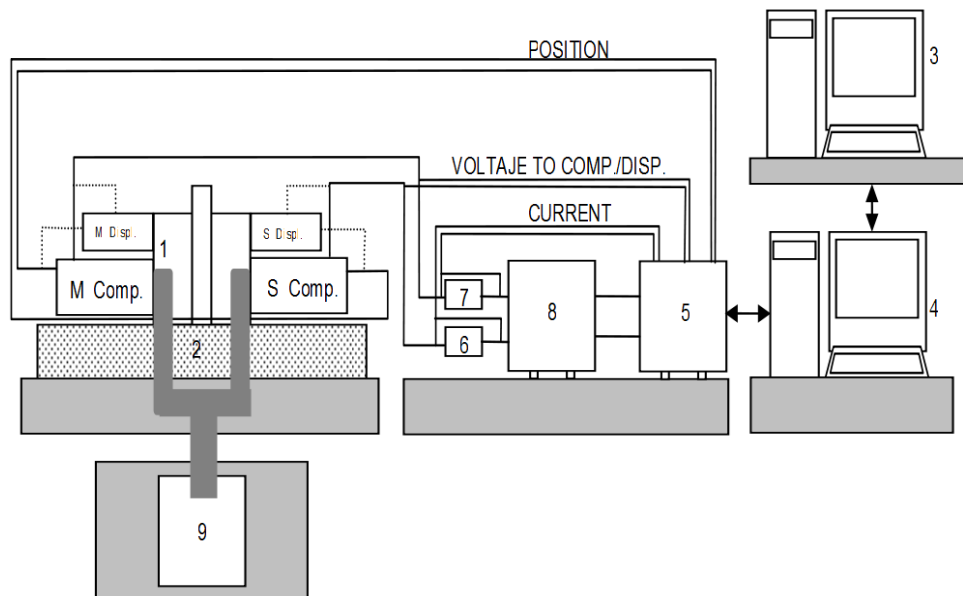


Fig. 9: Schematics of the test-rig built by CRISA

5. Further details of the implementation of the vibration-cancellation algorithm in the test-rig

The vibration-cancellation algorithm described in this document was proposed by Comillas-IIT to be used by CRISA in compressors for cryo-coolers. Comillas-IIT has assisted CRISA in the final stages of the implementation and in the experimental tests carried out. It is worth mentioning that,

- In the test-rig built by CRISA only one accelerometer is used for vibration feedback. Therefore, $n_s=1$ in (1).
- Fundamental component (1st harmonic) and harmonics from 2 to 7 have been considered for cancellation.
- The slave compressor command signal stops been updated when the amplitudes of all the harmonics considered are below the established limit. Since the optimization process is carried out for each harmonic individually, it is possible to establish a different limit for each harmonic in the actual implementation. In the prototype harmonic-amplitude limits between 10 μg and 350 mg are possible.
- The vibration-cancellation algorithm calculates real part and imaginary part of the command signal to be applied for each of the harmonics being considered. However, in the actual implementation, amplitude and phase of equivalent sinusoidal signals are calculated and used (one pair for each harmonic). The sinusoidal functions for all harmonics are then summed to generate the final command signal for the slave compressor. Therefore, it has been necessary to implement an algorithm to calculate the square root and another one to implement the arc-tangent in order to calculate amplitude and phase of a complex number, respectively. The implementation of the arc-tangent has been specially troublesome in order to achieve the required precision.
- In the prototype tested, the command signal for the slave compressor is updated always when there is at least one harmonic out of limits. The part of the command signal related with harmonics already within limits is also updated adding zero to its previous value. Unfortunately, this procedure requires the use of the arc-tangent function even for those harmonics within limits, causing systematic errors and small variations on all harmonic components of the command signal, including those that would not need updating.
- The vibration-cancellation algorithm is based on the approximate calculation of sensitivities and this is only reliable when small perturbations take place. Therefore, the command increment used in real and imaginary parts of all harmonics when calculating sensitivities must be programmed to a small value by the user using one of the parameters described in Section 3.5. With the sensitivities, the algorithm calculates the direction to increment the command signal for each harmonic.

- The size of the command-signal increment in the direction calculated using sensitivities is also limited. A small value for this limit makes the algorithm slow while a large value for this limit would speed-up convergence. However, a large value of this limit would also make the algorithm prone to convergence problems due to noise and non-linearities of the system. Since the largest vibration component will be at the fundamental frequency, the increment limit for this frequency has been set to a larger value than the limit for the harmonics. Increments of between 5 and 30 μm for the fundamental frequency component have been used successfully in the test-rig. This parameter can be set up to 4.5 mm in the prototype, however. The same maximum limit can be established for the harmonics but the experiments have shown that increments of 5 μm are sufficient.
- A limit for the amplitude of each harmonic in the command signal has also been established. In the prototype, this value can be up to 4.5mm and can be set for each harmonic independently. In the tests carried out in the test-rig, the amplitude of each harmonic in the command signal has always been less than 20 μm .
- In the prototype used by CRISA, command increments have been applied using a ramp of, typically, 4.5 mm/20s.
- The number of latency cycles (N_{lat}) can be set between 1 and 128, in the prototype. Since changes in the command signal are implemented gradually (with a ramp), the algorithm has to wait the time that the command signal reaches its required value plus the time to reach the steady state with the new command. Eventually, the number of latency cycles has been set to 64 in the prototype.
- The number of cycles to be used in the DFT can also vary between 1 and 128, in the prototype. The larger this number is the more selective the DFT is. Tests with 64 cycles per DFT have given good results with harmonic limits between, 1 mg and 10 mg.

6. Experimental results of the vibration-cancellation algorithm

6.1 Vibration-cancellation without temperature control

The vibration-cancellation algorithm has first been tested in the laboratory prototype without temperature control and the results are presented in Figs. 10 to 13. The experiments have been carried out under the following conditions:

- The cooler+CDE system has been commanded so that a stable temperature is reached in the cold-tip of the prototype. This has been achieved disconnecting the temperature control and generating a pre-established and constant-amplitude position reference for the compressor linear motors. In one of the linear motors

of the compressor, the position reference is modified by the vibration-cancellation algorithm (slave compressor).

- The amplitude for the master-compressor position reference has been set to 1.5mm. This produces 85K and 65K in the cold-tips.

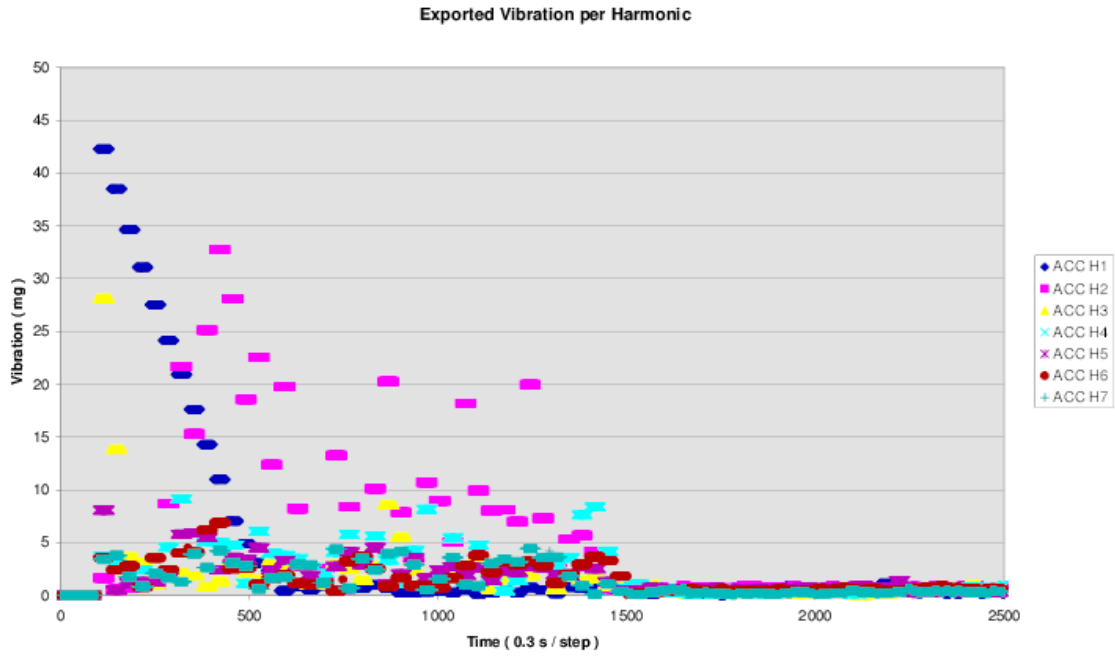


Fig. 10: Performance of the vibration-cancellation algorithm with 1mg of amplitude limit. Fast cancellation. Overview.

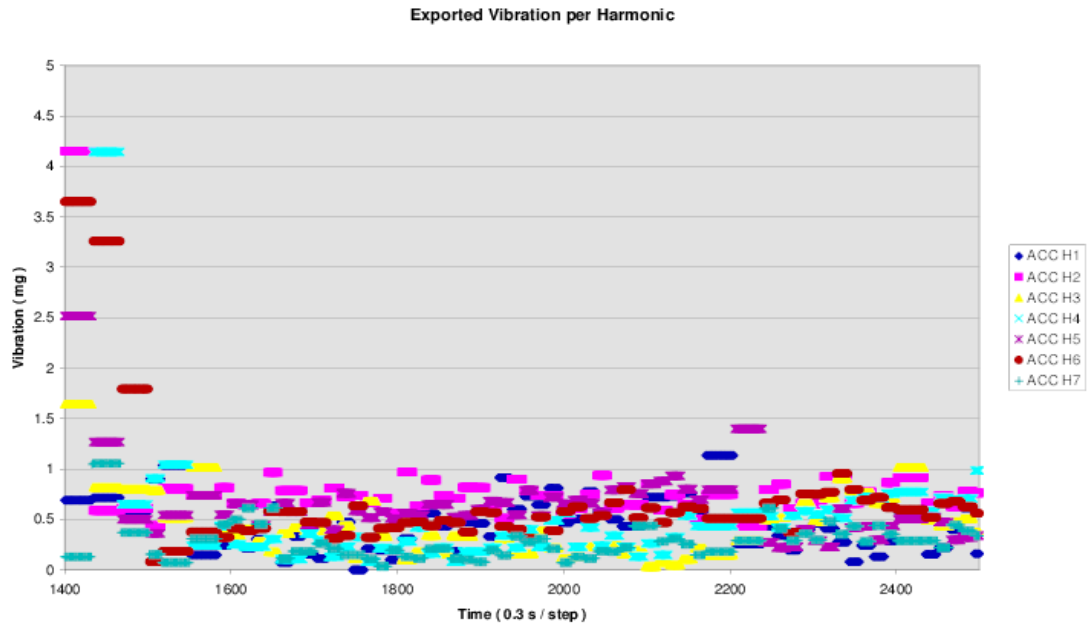


Fig. 11: Performance of the vibration-cancellation algorithm with 1mg of amplitude limit. Fast cancellation. Final stage.

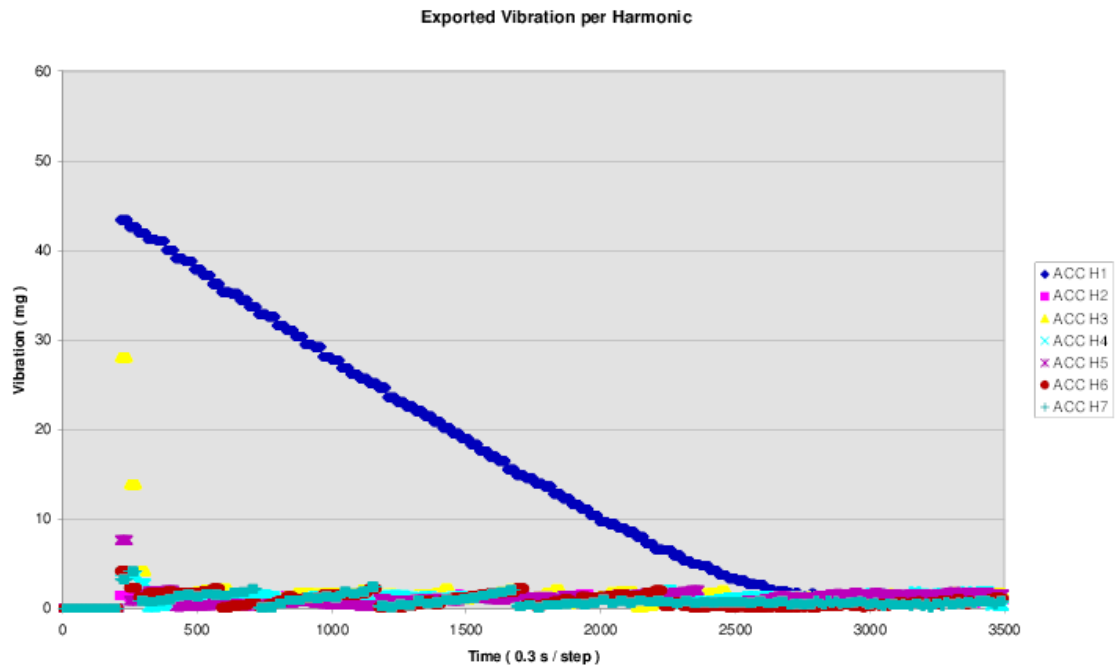


Fig. 12: Performance of the vibration-cancellation algorithm with 2mg of harmonic limit. Slow cancellation. Overview.

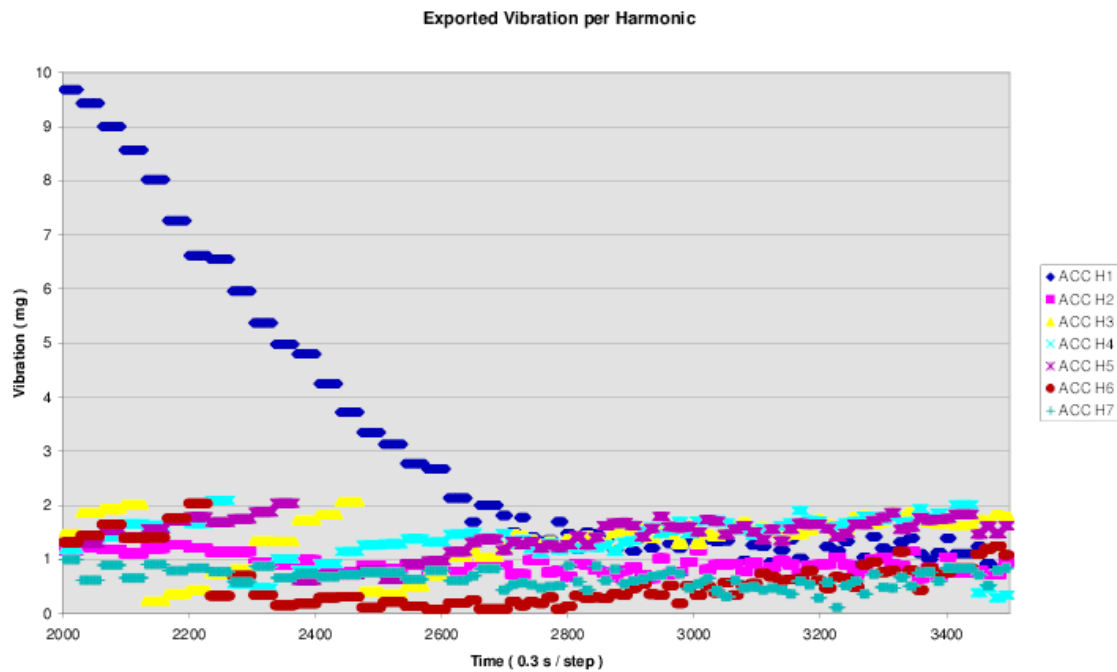


Fig. 13: Performance of the vibration-cancellation algorithm with a 2mg harmonic limit. Slow cancellation. Final stage.

6.2 Testing the vibration-cancellation algorithm with temperature control

The vibration-cancellation algorithm has also been tested in the laboratory when the temperature control was working. However, the tests have been performed in flight non-representative conditions. To start with, the test setup does not include a thermal load, and hence thermal inertia of the cold-tip was very small. In addition, the temperature sensor is a Cernox 1030 instead of the Cernox 1080, for which the CDE is designed and, therefore, temperature is acquired with much worse resolution than initially planned. Under these circumstances, the position reference for the master compressor is continuously varying and the vibration related to the fundamental (first) harmonic cannot be cancelled.

A detail of the results when vibration-cancellation and temperature-control algorithms were working simultaneously can be seen in Fig.14. The temperature reference was set to 80K and the harmonic limit was set to 3mg.

A second set of experiments was carried out removing the integral action of the temperature controller and making the control slower, in order to avoid having sharp changes of the demanded master-compressor stroke. In these cases, the vibration-cancellation algorithm succeeds cancelling all the harmonics.

Figures 15 and 16 show a situation in which the temperature controller does not disturb the vibration-cancellation algorithm (during cooling). It is shown that all harmonics can be reduced under 1mg.

Finally, Figures 17 and 18 show the results when only one of the compressors is working at the beginning (CDE). Again all vibration frequencies but the fundamental one can be driven under the limit of 1mg.

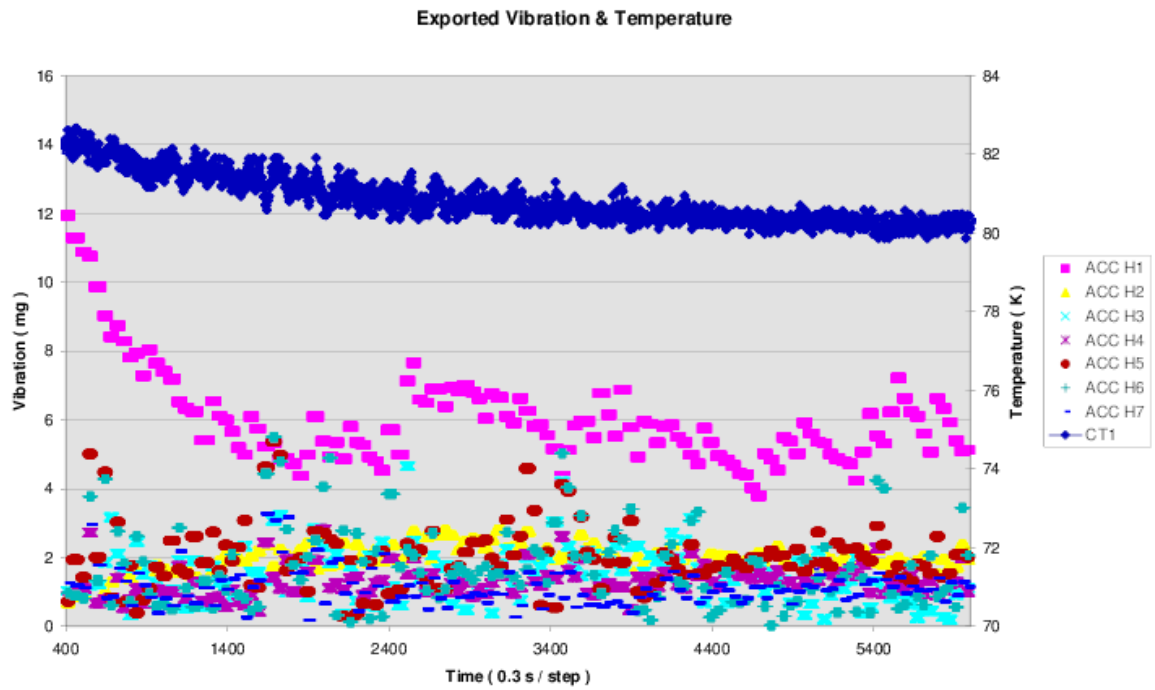


Fig. 14: Experimental results with vibration-cancellation and temperature control working simultaneously

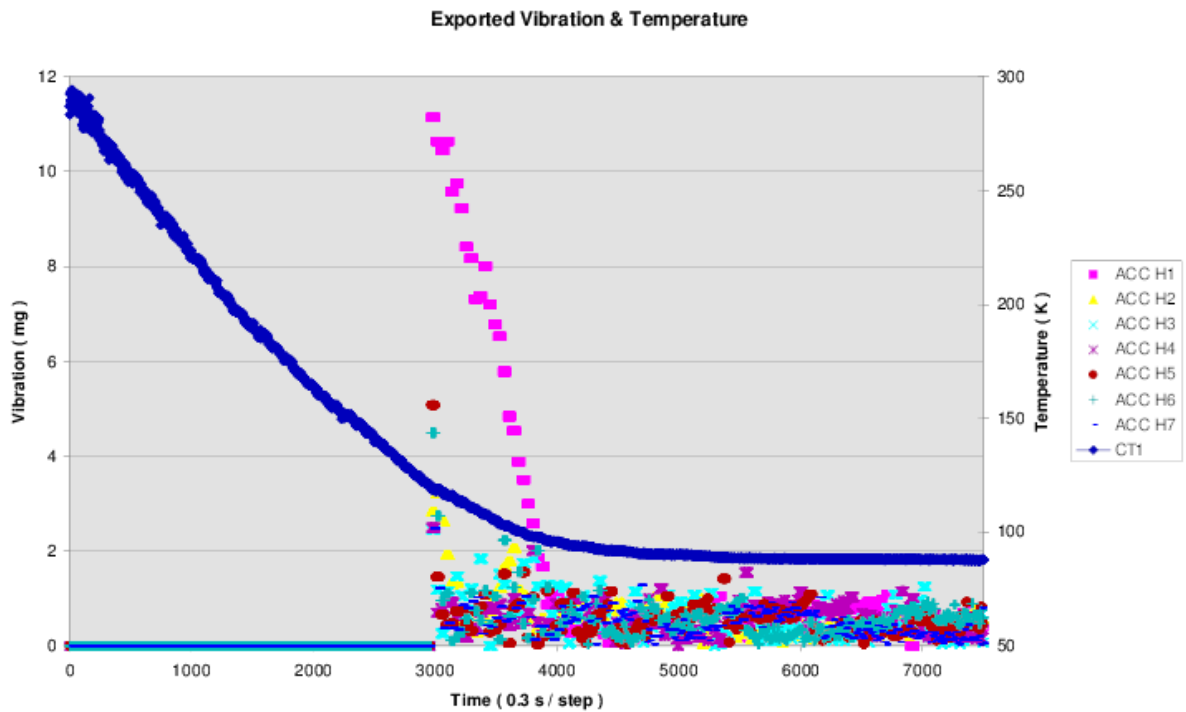


Fig. 15: Experimental results. Cooling when temperature control and vibration-cancellation algorithm are working simultaneously. Overview

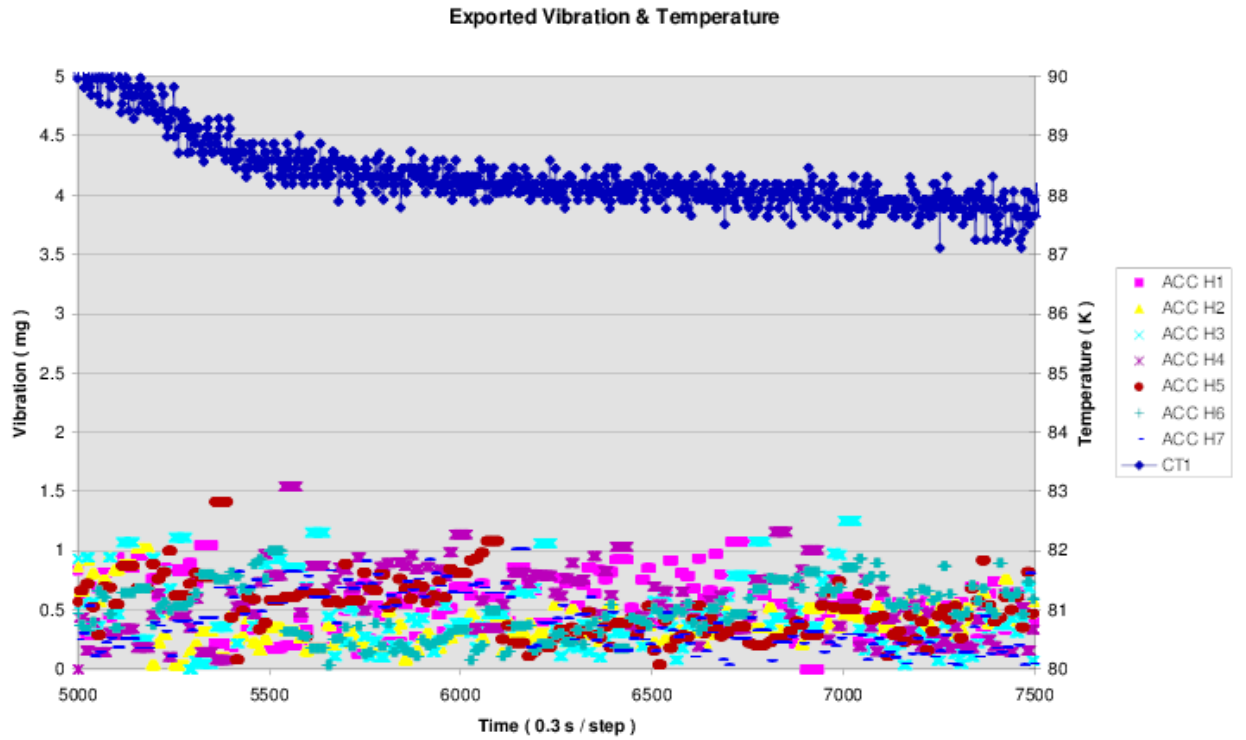


Fig. 16: Experimental results. Cooling when temperature control and vibration-cancellation algorithm are working simultaneously. Detail.

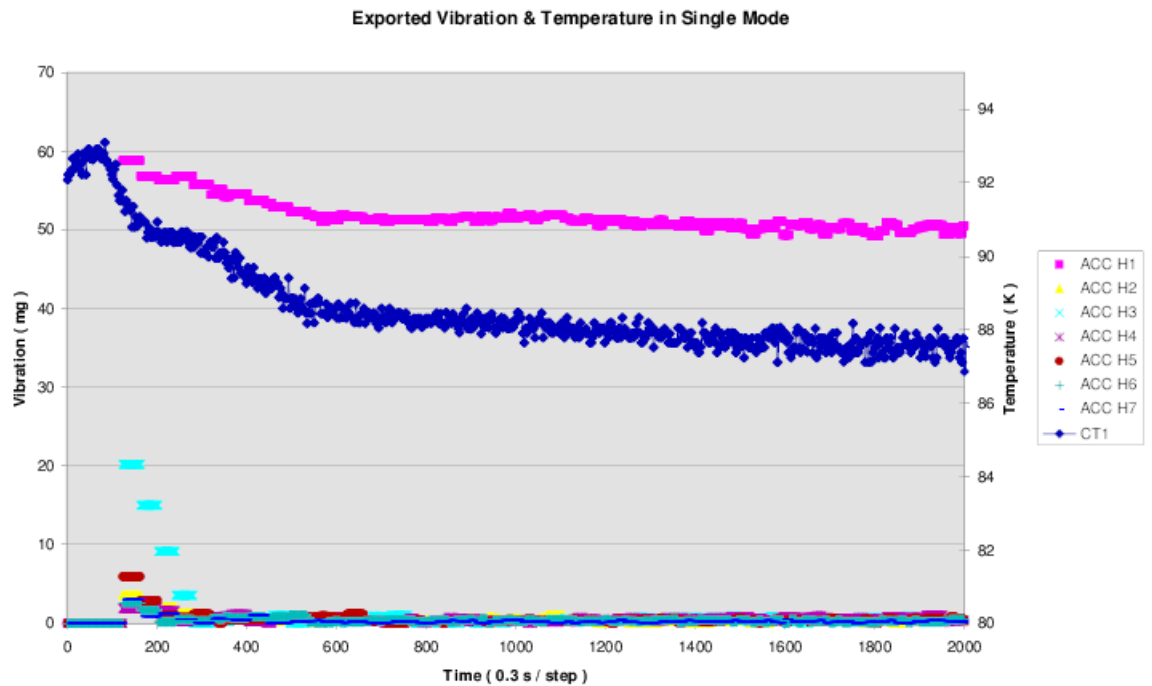


Fig. 17: Experimental results. Cooling when temperature control and vibration-cancellation algorithm are working simultaneously. Only one of the compressors is working. Overview

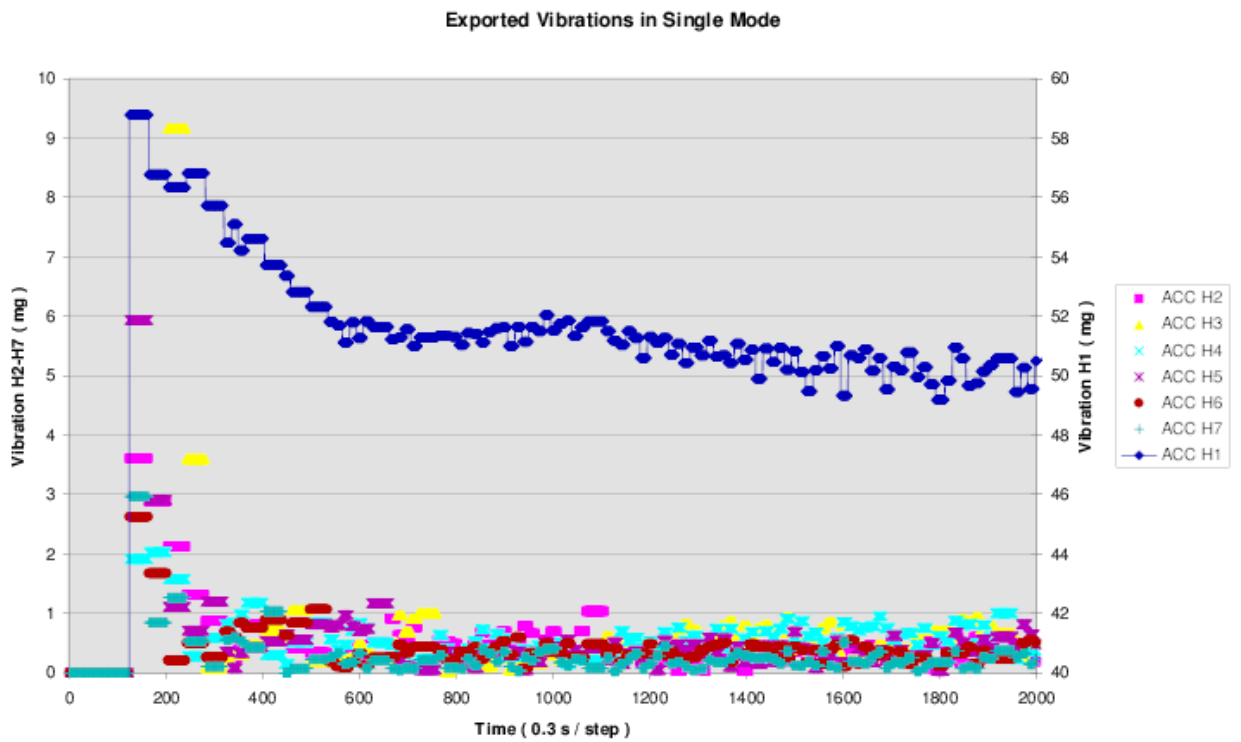


Fig. 18: Experimental results. Cooling when temperature control and vibration-cancellation algorithm are working simultaneously. Only one of the compressors is working. Detail